



Universidad  
Carlos III de Madrid

Departamento de Informática

PROYECTO FIN DE CARRERA

# SISTEMA DE GESTIÓN DE CARTERA DE PROYECTOS INFORMÁTICOS

Autor: Pablo Rincón Rodríguez

Tutor: José Luis López Cuadrado

Leganés, Enero de 2015



Título: Sistema de Gestión de Cartera de Proyectos Informáticos

Autor: Pablo Rincón Rodríguez

Director: José Luis López Cuadrado

## EL TRIBUNAL

Presidente: \_\_\_\_\_

Vocal: \_\_\_\_\_

Secretario: \_\_\_\_\_

Realizado el acto de defensa y lectura del Proyecto Fin de Carrera el día \_\_\_\_ de Enero de 2015 en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de

VOCAL

SECRETARIO

PRESIDENTE



# Agradecimientos

Probablemente esta se la parte más complicada de todo el proyecto. Existen tantas y tantas personas que me han ayudado de una manera u otra a lo largo de todos estos años, que resulta complicado querer hablar de una sin que te lleve a otra y a otra.

En primer lugar, por su puesto se encuentra mi madre. Por desgracia, no pudo verme crecer, abandonar el colegio, entrar en la universidad, empezar a trabajar, y tantas y tantas cosas que la habrían colmado de felicidad, pero no por ello no ha dejado de pasar un solo día en el que no la recuerde y me anime a superar todas y cada una de las barreras que se me presentan a lo largo de mi vida. El tiempo que pudiste estar junto a mí me enseñaste lo importante que son los estudios y la cultura, y sin duda, sin ti no podría haberlo conseguido. !!!MUCHÍSIMAS GRACIAS MAMÀ!!!

Por otro lado, y no menos importantes, mi padre y mi hermano ,los cuales me han ido marcando a lo largo de los años el camino a seguir para que conseguir que a día de hoy pueda estar donde estoy hoy. Muchas gracias por vuestro apoyo constante.

A Fernando Paniagua, por comenzar conmigo este proyecto, ofreciéndome su punto de vista, dirigiendo el camino que debía de tener, y sobre todo por no abandonarme en ningún momento, incluso cuando no tenía necesidad, ni obligación ninguna de hacerlo. Por supuesto, este proyecto se debe en gran parte a tí!!

A José Luis López, por aceptarme con un proyecto ya comenzado, en el que se ha implicado totalmente para ofrecerme la oportunidad de poder finalizarlo y poder conseguir el tan ansiado título.

A cuanta más gente podría agradecer su apoyo y ayuda a lo largo de todos estos años, profesores, amigos, compañeros,.... gracias a todos y espero que sigamos estando juntos muchos años más.



# Resumen

Uno de los puntos más importantes para que la organización de una empresa funcione con total garantía es la gestión y comunicación interna. Sin comunicación es imposible que todos los miembros de la organización vayan en la misma dirección.

El proyecto que aquí se presenta trata de dar una solución a este problema, con el fin principal de ahorrar costes de gestión, mejorar la eficiencia de la empresa, mejorar el resultado y por tanto satisfacción del cliente, y de esta manera situar a la empresa en una posición de ventaja competitiva con respecto a sus competidores en el sector, permitiendo aumentar los ingresos y de esta manera poder crecer y expandirse

La solución propuesta se basa en la creación de una herramienta web, , con distintos niveles de funcionalidades disponibles por cada perfil de empleado, y en la que toda la información de los distintos proyectos de la empresa pueda ser obtenida por todos ellos, y de esta manera poder conocer de primera mano la situación de los mismos, descargarse documentación asociada, conocer fases esfuerzos asignados, etc. Por otro lado, el empleado podrá gestionar distintas opciones personales como solicitudes de vacaciones, ausencias, realizar el reporte de horas diario de una forma rápida y sencilla, etc.

Mediante sistemas de alerta automática, los empleados recibirán distintas alarmas creadas por sus responsables, en la que se les indicará información relevante para su día a día, como son fechas de comienzo de cada, fase, fechas de fin, etc.

Por otro lado, los responsables de la compañía tendrán la capacidad de realizar cualquier tipo de informe con toda la información introducida en la aplicación, pudiendo de esta manera estudiar el estado actual de la compañía, pudiendo dar una respuesta rápida a los problemas que surjan , e incluso poder anticiparse a ellos.

Finalmente, y de cara a que los distintos proyectos sean siempre realizados por el personal más cualificado para cada uno de ellos, de entre los distintos integrantes de la compañía, se define un sistema de autoasignación de recursos a un proyecto, mediante la conmutación de distintos pesos como son disponibilidad en las fechas de asignación, conocimientos de las tecnologías, evaluaciones recibidas, etc.

De esta manera, mediante un sistema de gestión de cartera y recursos adecuado lograremos una mayor eficacia de todos los componentes de la misma, llevando a la empresa a un posición de ventaja en su campo.

# Abstract

One of the most important points for the organization of a company, is to work with full warranty management and internal communication. Without communication, is impossible for all organization's members to work in the same direction.

The project that here is present, tries to give a solution to this problem, with the principal purpose of saving management's costs, improving the efficiency of the company, improving the result and therefore satisfaction of the client, and hereby to place the company in a position of competitive advantage with regard to his competitors in the sector, allowing to increase the income and hereby to be able to grow and expand

The proposed solution is based on the creation of a web-based tool accessible by all company employees with different levels of functionality available for each employee profile, and in which all the various projects information can be obtained for them all, and in this way to see first hand the situation of it, associated documentation, know phases assigned efforts, soon. In addition, the employee may manage different personal options as vacation requests, absences, performed daily reporting hours quickly and easily, etc.

Through automatic alert systems, employees receive different alarms created by their responsables, which will be told information relevant to their day, such as start dates of each, phase, end dates, etc.

Moreover, the company responsible will have the ability to make any kind of report with all the information stored in the application, and can thus study the current state of the company and provide a rapid response to emerging issues, and even to anticipate them.

Finally, and in view of the different projects are always made by the most qualified employee, from among the members of the company, a functionality of self-assignment of resources to a project is defined, by switching different weights such as availability at the time of assignment, knowledge of technologies, received evaluations, soon.

This way, through a system of portfolio management and adequate resources assignment, we can achieve greater efficiency of all components of the company, leading it to a position of advantage in its working field.





# Índice general

1 - Presentación del Proyecto.....	2
1.1 - Introducción.....	2
1.2 - Motivación del Proyecto.....	3
1.3 - Objetivos del Proyecto.....	4
1.4 - Alcance Funcional .....	5
1.5 - Medios empleados en la solución.....	7
1.6 - Contenido de la memoria.....	8
2 - Estado del Arte.....	10
2.1 - Herramientas de Desarrollo:.....	10
2.2 - Herramientas de Almacenamiento.....	16
2.3 - Herramientas de Hospedaje / Servidor .....	17
2.4 - Arquitectura .....	20
2.5 - Conclusiones al estado del arte.....	21
3 - Estudio Previo y Definición de Requisitos.....	22
3.1 - Introducción.....	22
3.2 - Fase de Estudio Previo.....	23
3.3 - Requisitos de Usuario .....	25
4 - Estudio Analítico del Proyecto .....	32
4.1 - Ciclo de vida.....	32
4.2 - Organigrama y Recursos Humanos.....	35
5 - Diseño de la Aplicación .....	41
5.1 - Introducción.....	41
5.2 - Casos de Uso.....	42
5.2.2 - Diagrama de los Casos de Uso .....	43
5.3 - Modelo de Datos: .....	66

5.4 - Modelo Entidad/Relación de la BBDD .....	81
5.5 - Modelo Relacional de la BBDD.....	83
5.6 - Esquema de BBDD .....	86
5.7 - Diseño General del Diagramas de Clases .....	89
5.8 - Diseño del proceso de ejecución “Demonios” .....	96
5.9 - Diagramas de secuencia .....	101
6 - Codificación.....	106
6.1 - Estructura de proyecto Java .....	107
6.2 - Normas y consejos de codificación.....	114
6.3 - Librerías externas utilizadas dentro de la aplicación.....	117
7 - Pruebas de Sistema .....	119
8 - Despliegue .....	127
8.1 - Creación de nuevo servidor .....	128
8.2 - Configuración del servidor .....	131
8.3 - Despliegue de la Aplicación.....	135
8.4 - Creación de base de datos, tablas y datos .....	137
8.5 - Configuración de directorios .....	140
9 - Navegabilidad y Guía de Usuario .....	141
9.1- Navegabilidad.....	142
9.2 - Guía de Usuario .....	146
10 - Presupuesto .....	147
10.1 - Recursos Humanos .....	147
10.2 - Recursos Materiales.....	148
10.3 - Gastos Indirectos.....	150
10.4 - Presupuesto Final.....	151
11 - Conclusiones y líneas futuras.....	153
11.1 - Conclusiones.....	154
11.1 - Líneas Futuras.....	156

12 - Anexos .....	157
12.1 - Anexo Pruebas Sistema Gestion Cartera .....	157
12.2 - Anexo Manual de Usuario .....	157
12.3 - Anexo Presupuesto de Proyecto .....	158
12.4 - Anexo preparación base de datos. ....	158
13 - Bibliografía .....	162
13.1 - Libros .....	162
13.1 - Páginas Web .....	163

# Índice de figuras

Figura 1. - Ciclo de Vida de un proyecto .....	33
Figura 2. - Organigrama de Recursos Humanos .....	37
Figura 3 - Planificación general del proyecto.....	38
Figura 4 - Diagrama de Gantt del proyecto.....	39
Figura 5 - Porcentajes de tiempo por fase .....	39
Figura 6 - Diagrama de casos de uso de la aplicación .....	43
Figura 7 - DLL tabla Alarmas.....	66
Figura 8 - DLL tabla Calendar .....	67
Figura 9 - DLL tabla Cat_Condiciones_Alarmas .....	67
<i>Figura 10 - DLL tabla Cat_Equipos .....</i>	<i>67</i>
Figura 11 - DLL tabla Cat_Fases.....	68
Figura 12 - DLL tabla Cat_Herramientas.....	68
Figura 13 - DLL tabla Cat_Releases .....	68
Figura 14 - DLL tabla Cat_Tipos_Alarmas .....	69
Figura 15 - DLL tabla Cat_Condiciones_Alarmas.....	69
Figura 16 - DLL tabla Conocimientos_Proyecto.....	70
Figura 17 - DLL tabla Derechos.....	71
Figura 18 - DLL tabla Empleados .....	73
Figura 19 - DLL tabla Funcionalidad_Pagina.....	74
Figura 20 - DLL tabla Gestion_Errores .....	74
Figura 21 - DLL tabla Gestion_OK .....	74
Figura 22 - DLL tabla Informes.....	75
Figura 23 - DLL tabla Login .....	75
Figura 24 - DLL tabla Mensajes_Sistema .....	76

Figura 25 - DLL tabla Planificacion_Informes.....	77
Figura 26 - DLL tabla Planificacion_Proyecto.....	77
Figura 27 - DLL tabla Proyecto .....	78
Figura 28 - DLL tabla Rel_Proyecto_Empleado .....	79
Figura 29 - DLL tabla Reporte_Horas .....	79
Figura 30 - DLL tabla Solicitudes.....	80
Figura 31 - Grafica Diagrama Entidad Relación.....	81
Figura 32 - Grafica Diagrama Entidad Relación (cont.) .....	82
Figura 33 - Diagrama Relacional de Base de Datos .....	83
Figura 34 - Diagrama Relacional de Base de Datos (cont.) .....	84
Figura 35 - Diagrama Relacional de Base de Datos (cont.2).....	85
Figura 36 - Esquema de Base de Datos.....	86
Figura 37 - Esquema de Base de Datos (cont.) .....	87
Figura 38 - Esquema de Base de Datos (cont.2) .....	87
Figura 39 - Diagrama de clases general.....	89
Figura 40 - Diagrama arquitectónico de alto nivel.....	90
Figura 41 - Diagrama clases Web vs Servlet.....	90
Figura 42 - Diagrama clases Servlet vs Gestor .....	91
Figura 43 - Diagrama clases Gestor vs Modelológico .....	91
Figura 44 - Diagrama clases Gestor vs Modelológico ( cont.) .....	92
Figura 45 - Diagrama clases Gestor vs Modelológico ( cont.2) .....	92
Figura 46 - Diagrama clases Gestor vs Persistencia.....	93
Figura 47 - Diagrama clases Gestor vs Persistencia (cont.) .....	93
Figura 48 - Diagrama clases Gestor vs Persistencia (cont.2) .....	94
Figura 49 - Diagrama clases Gestor vs Persistencia (cont.3) .....	94
Figura 50 - Diagrama clases paquete mail.....	95
Figura 51 - Diagrama demonios.....	95
Figura 52 - Lanzadores demonios .....	95

Figura 53 - Diagrama de Flujo de Demonio de Alarmas .....	98
Figura 54 - Diagrama de Flujo de Demonio de Informes .....	100
Figura 55 - Diagrama Secuencia creación de proyecto .....	102
Figura 56 - Diagrama Secuencia creación de alarma .....	102
Figura 57 - Diagrama Secuencia creación de nuevo informe .....	103
Figura 58 - Diagrama Secuencia creación de nuevo empleado .....	104
Figura 59 - Diagrama Secuencia autoasignación de proyecto .....	105
Figura 60 - Estructura de proyecto generada .....	107
Figura 61 - Estructura de carpetas de proyecto generado .....	108
<b>Figura 62</b> - Estructura de carpetas de parte WEB .....	109
Figura 63 - Detalle paquetes creados para el proyecto .....	110
Figura 64 - Detalle demonios creados para el proyecto.....	110
Figura 65 - Detalle gestores creados para el proyecto.....	111
Figura 66 - Detalle paquete mail.....	111
Figura 67 - Detalle paquete modelológico generado en el proyecto.....	112
Figura 68 - Detalle paquete persistencia generado en el proyecto.....	112
Figura 69- Detalle de servlets generados en el proyecto.....	113
Figura 70 - Lanzadores de los demonios del alarmas e informes.....	113
Figura 71 - Estructura principal de la parte web de la aplicación .....	116
Figura 72 - Ejemplo visualización de datos con Dtree .....	118
Figura 73 - Ejemplo visualización de datos con jquery y datatables.....	118
Figura 74 - Ejemplo visualización de datos con jscalendar .....	118
Figura 75- Pagina de selección de servicio en AWS.....	128
Figura 76 - Selección de localización de servidor en AWS .....	128
Figura 77 - Creación de nueva instancia de servicio en AWS .....	129
Figura 78 - Selección de sistema operativo en AWS .....	129
Figura 79 - Selección de tipo de servidor en AWS .....	129
Figura 80 - Pagina de correcta creación de servidor en AWS .....	130

Figura 81 - Instancia correctamente creada en AWS.....	130
Figura 82 - Apertura de puertos en AWS .....	131
Figura 83 - Conexión mediante putty al nuevo servidor mediante sftp.....	131
Figura 84 - Actualización de funciones principales en el servidor .....	132
Figura 85 - Actualización de funciones principales en el servidor .....	132
Figura 86 - Comando de selección de paquetes a instalar .....	132
Figura 87 - Selección de aplicaciones a instalar dentro del servidor .....	133
Figura 88 - Inserción de password para usuario root de Mysql .....	133
Figura 89 - Tabla de pivotaje para redireccionamiento de puertos .....	133
Figura 90 - Inclusión de usuario y password de tomcat .....	134
Figura 91 - Actualización de administrador tomcat .....	134
Figura 92 - Reinicio de aplicación tomcat.....	134
Figura 93 - Pagina de correcta instalación de tomcat .....	135
Figura 94 - Autenticación en Tomcat .....	135
Figura 95 - Despliegue de proyecto war en Tomcat.....	136
Figura 96 - Correcto despliegue de la aplicación en el servidor .....	136
Figura 97 - Conexión a Mysql.....	137
Figura 98 - Creación de Base de Datos con Mysql .....	137
Figura 99 - Cambio de Base de Datos en Mysql.....	137
Figura 100 - Exportación de datos desde Mysql en desarrollo .....	138
Figura 101 - Sqls generadas en la exportación de Mysql.....	138
Figura 102 - Subida de ficheros mediante Filezilla .....	139
Figura 103 - Concatenación de archivos de sql .....	139
Figura 104 - Carga de archivo en el entorno de producción con mysql.....	139
Figura 105 - Creación de directorios de informes y documentación en servidor .....	140
Figura 106 - Cambio de permisos de directorios de informes y documentación en servidor.....	140
Figura 107 - Menú de navegación de aplicación.....	142



Figura 108 - Navegación de Menú Cartera.....	143
Figura 109- Navegación de Menú Empleado .....	143
Figura 110 - Navegación de Menú Alarmas.....	144
Figura 111 - Navegación de Menú Informes.....	144
Figura 112 - Navegación de Menú Administración .....	145
Figura 113 - Navegación de Mensajería Sistema.....	145
Figura 114 - Coste Hora de Recurso .....	147
Figura 115 - Coste Total pro Recurso.....	147
Figura 116 - Requisitos de servidor y disco .....	149
Figura 117 - Tarifa tri-anual de coste de servidor .....	149
Figura 118 - Coste por Recurso.....	151
Figura 119 - Coste Total del Proyecto .....	152



# 1 - Presentación del Proyecto

## 1.1 - Introducción

Este proyecto surge no solo con la intención de la realización de un proyecto de final de carrera que certifique mis estudios y me otorgue de esta manera el título de finalización de la carrera, sino que dada la experiencia profesional que he ido obteniendo en los últimos años, he tratado de realizar una aplicación que sea capaz de solventar varias de las importantes carencias que actualmente me he encontrado, tratando así de aumentar la productividad, calidad, y beneficios que se obtienen y se ofrecen por proyecto al usuario, con el simple hecho de aprovechar al máximo los recursos que la empresa ya posee, y estableciendo una herramienta única de comunicación entre todas las secciones de la misma.

A lo largo de este documento se podrá observar el avance del ciclo de vida de un proyecto partiendo de la definición de necesidades, establecimiento de requisitos, etc., hasta su puesta final en productivo.

## 1.2 - Motivación del Proyecto

Tras la experiencia adquirida, y los distintos problemas que me he ido encontrando a lo largo de estos años, una de las carencias más importantes que he detectado, y la cual ha sido una gran sorpresa para mí, ha sido ver y experimentar como la herramienta más utilizada dentro del mundo de la informática para todo tipo de requerimientos, continúa siendo la herramienta "Excel de Microsoft". Efectivamente es una herramienta que te ofrece una gran capacidad de realizar una cantidad de tareas, que son difíciles de igualar por otros productos, pero no por ello es la solución ideal para todo tipo de requisitos y necesidades.

Dentro de algunas de las empresas con las que he ido colaborando a lo largo de estos años, esta herramienta es utilizada tanto para establecer los requisitos del usuario, planificaciones, estimaciones, imputaciones, asignaciones de tareas, estudios de productividad, etc. , de forma unitaria.

Esto sin duda nos lleva a que la información se encuentra repartida en distintos documentos, PC's, con difícil acceso, lo que hace que en la mayor parte de las ocasiones, ni siquiera se conozca su existencia, e incluso lleve a dar un servicio al cliente bastante deficiente, no por incapacidad de la empresa, sino por desconocimiento de la realidad existente.

De esta manera, surgió la idea de la realización de una aplicación que permitiese en sí misma llevar el control de los productos/proyectos existentes, tanto para los responsables, del mismo, como para los desarrolladores, recursos humanos, y resto de eslabones de la empresa, que junto con una asignación "inteligente" de recursos, y la capacidad de generación de informes, le proporcione a la empresa un valor añadido sobre las demás, pudiendo rebajar costes, aumentando la productividad, y de esta manera pudiendo tener un ventaja competitiva dentro del mercado actual.

Es cierto, que existen distintas herramientas, en el mercado que te ofrecen capacidades similares, las cuales todas juntas formarían una funcionalidad semejante. Pero en este punto surge la pregunta de, ¿No me saldría más rentable a la larga, desarrollar una aplicación que cubra exactamente las necesidades de la compañía, pudiendo desarrollarla por las ramas más convenientes para la misma, que pagar licencias excesivamente caras a terceros, por productos que pueden no llegar a cumplir nuestra necesidad?.

## 1.3 - Objetivos del Proyecto

Dadas las motivaciones del apartado 1.2, dentro de este proyecto se parte con un objetivo claro, que se basa en la realización completa de un ciclo de vida de un proyecto, partiendo desde el primer contacto con los usuarios, hasta llegar a la solución más óptima para estos, y que cubra las necesidades que la compañía tenga.

De esta manera, este proyecto parte de la base de la existencia de una empresa ficticia denominada "**Vellés System Consulting S.L**" ( **V.S.C.** en adelante, la cual está dedicada a la consultoría informática, pero sin capacidad actual de llevar adelante este tipo de proyectos de forma autónoma por no el tener el "know-how" en este tipo de estudios y proyectos), la cual solicitará los servicios de la compañía "**PRR Business Solution S.L.**" ( **P.R.R.B.S.** en adelante, la cual es una compañía especializada en el estudio, desarrollo e implantación de soluciones a empresas), al detectar un problema grave de falta de comunicación interno dentro de su compañía, con el objetivo estudiar todos y cada uno de los defectos y problemas de la empresa y dar una solución global a todos ellos.

La solución de este proyecto describirá todas las fases por las que el proyecto debería de pasar, las cuales a alto nivel serían las siguientes:

- Toma de contacto, estudio y definición de requisitos del proyecto
- Planteamiento de solución y planificación acorde a los requisitos planteados
- Diseño del Proyecto
- Desarrollo y Pruebas de la solución
- Formación a los futuros usuarios del uso de la aplicación
- Despliegue en productivo de la aplicación
- Monitorización y evaluación Final de la solución presentada.

## 1.4 - Alcance Funcional

Como ya se ha ido introduciendo en los dos capítulos anteriores, el alcance funcional del proyecto engloba todo el ciclo de vida del proyecto desde la primera toma de contacto, hasta el establecimiento del producto final en productivo.

A forma de introducción de la solución que finalmente será tomada dentro del proyecto, se propondrá y llevará a cabo una aplicación de gestión integrada que englobe la información básica de todos los sectores de la empresa, y que permita la comunicación mediante una única vía entre todos estos sectores, de manera que se consiga:

- Situar a la empresa en una posición de ventaja competitiva con el resto de empresas que trabajan dentro de su mismo mercado de negocio, debido a la agilidad y eficacia de la empresa.
- Ahorrar costes por pagos de licencias a terceros.
- Capacidad de ahorro de tiempo y esfuerzo que llevan a cabo gran parte de las tareas de gestión dentro de la empresa, lo que permitirá generar ofertas más ajustadas, o incrementar el beneficio neto por producto.
- Unificación de todas las herramientas internas de planificación, gestión, comunicación, de manera que toda la información de la empresa esté disponible para todos los empleados autorizados, etc.
- Capacidad de analizar los datos de los distintos proyectos de la empresa, pudiendo de esta manera adelantarse a posibles problemas.

Para conseguirlo, vamos a partir del punto de vista global de la empresa, identificando cual es su necesidad, que es aquello que puede convertir a la empresa, en pionera en su ámbito, y a partir de ahí, iremos recorriendo los distintos ciclos de vida del producto, hasta conseguir implantar el mismo en un entorno productivo donde este sea por fin una herramienta indispensable dentro de la empresa.

Las principales características que nos aportará la solución serán:

- Conseguir comunicar a todos los departamentos de la empresa.
- Obtención de informes de productividad, eficacia, etc.
- Sistema de gestión de alertas
- Asignación efectiva y automática de recursos
- Unificación de la información de la empresa
- Interfaz Web, y con acceso por todos los empleados
- La información estará disponible según perfiles.
- La información se plantea pueda ser accedida mediante un explorador de internet, dentro del cual, tras pasar el proceso de autenticación, nos permita explotar la información existente.

La funcionalidad inicial que se le pretender dar a esta aplicación es la siguiente:

- Interfaz gráfica web, fácil e intuitiva
- Perfiles de acceso según el rol de cada empleado
- Sistema de gestión de productos
- Sistema de gestión de empleados
- Sistema de asignación automática de recurso
- Sistema de Gestión de alertas.
- Herramienta de generación de informes

## 1.5 - Medios empleados en la solución

Dado que con la solución tomada se trataba de dar una respuesta a los problemas de una pequeña compañía con un margen de beneficios no muy elevado, se ha considerado como necesidad prioritaria la utilización de elementos bien gratuitos, o con el menor coste de utilización y mantenimiento posible .

De esta manera de cara a la construcción del sistema se utilizarán los siguientes medios:

- **NetBeans** --> Entorno de desarrollo de código libre y coste 0.
- **Java, JavaScript, Html** --> Lenguajes de programación gratuitos,
- **Mysql** --> Sistema de gestión de base de datos relacional, de software libre, con un coste de 2000 dólares anuales.
- **Servidor AWS** --> Servidor externo perteneciente a Amazon, con un coste inferior a un servidor propio, al evitarnos servicios de mantenimiento, backups, etc.
- **Apache Tomcat7** --> Herramienta de despliegue de herramientas basadas en Java. Incluido en la suscripción a AWS.
- **Sistema operativo Ubuntu** --> Sistema operativo incluido en la suscripción a AWS.



## 1.6 - Contenido de la memoria

De cara a la mejor comprensión del documento de memoria del proyecto de Gestión de Cartera. dentro de este apartado se detallan los distintos puntos que se tratan en la misma, estructura y composición de forma general.

Estos puntos son:

- Presentación del proyecto: Descripción del proyecto tratado, indicando los motivos por los que ha sido elegido el mismo, descripción de la situación ficticia que se lleva a cabo, alcance de la herramienta que se va a desarrollar, presentado la problemática y la intención de solución para la realización del proyecto. Tratado en los apartados anteriores.
- Estado del Arte: Descripción y explicación de los distintos conceptos que van a ser tratados dentro del proyecto, y que son imprescindibles para un correcto entendimiento del mismo, incluidos las herramientas y recursos utilizados para la realización del mismo.
- Estudio analítico del proyecto: estudio previo de la situación actual en la que se encuentra las distintas funcionalidades parejas a este proyecto, y la base por la cual se empezará a desarrollar la herramienta de software.
- Diseño de la aplicación: apartado que contendrá todos los pasos de diseño necesarios para la generación de la herramienta o aplicación.
- Codificación: estudio de los estándares de codificación de scripts y estructura de los documentos.
- Pruebas: se expondrá una batería de pruebas generadas con la herramienta desarrollada con el fin de validar el buen funcionamiento de la aplicación.
- Navegabilidad e Interfaces de la aplicación: se especificará los diagramas de navegabilidad de los distintos tipos de usuarios, así como los interfaces gráficos de cada usuario, que servirán además como mini guías de orientación.
- Presupuesto: Se detallarán los costes que conlleva la realización de la aplicación, estableciendo de esta manera el precio más bajo posible por el que el mismo pueda ser vendido.

- Conclusiones y líneas futuras: determinaciones a las que se ha llegado tras el estudio de los resultados que genera la aplicación desarrollada y estudio de las líneas a las que puede evolucionar la herramienta en un futuro.
- Bibliografía: relación de fuentes documentales utilizadas para la redacción de este documento, teniendo en cuenta como fuentes documentales, los libros, trabajos o recursos obtenidos de Internet, de forma que se pueda consultar o profundizar en la información que en este documento se trata.

## 2 - Estado del Arte

### 2.1 - Herramientas de Desarrollo:

#### 2.1.1 - HTML:

Lenguaje muy sencillo que permite describir hipertexto, es decir, texto presentado de forma estructurada y agradable, con enlaces que conducen a otros documentos o fuentes de información relacionadas, y con inserciones multimedia (gráficos, sonido...).

La descripción se basa en especificar en el texto la estructura lógica del contenido (títulos, párrafos de texto normal, enumeraciones, definiciones, citas, etc.) así como los diferentes efectos que se quieren dar (especificar los lugares del documento donde se debe poner cursiva, negrita, o un gráfico determinado) y dejar que luego la presentación final de dicho hipertexto se realice por un programa especializado.

#### 2.1.2 - JSP:

Un JSP se compila a un programa en Java la primera vez que se invoca, y del programa en Java se crea una clase que se empieza a ejecutar en el servidor como un servlet.

La principal diferencia entre los servlets y los JSPs es el enfoque de la programación: un JSP es una página Web con etiquetas especiales y código Java incrustado, mientras que un servlet es un programa que recibe peticiones y genera a partir de ellas una página web.

Ambos necesitan un programa que los contenga, y sea el que envíe efectivamente páginas web al servidor, y reciba las peticiones, las distribuya entre los servlets, y lleve a cabo todas las tareas de gestión propias de un servidor web. Mientras que servidores como el Apache están especialmente pensados para páginas web estáticas con CGI, y programas ejecutados por el servidor, tales como pueden ser el PHP, hay otros servidores específicos para servlets y JSPs llamados contenedores de servlets o servlet engines.

### 2.1.3 - Java:

Java es un lenguaje de programación orientado a objetos y desarrollado por Sun Microsystems. Se trata de un entorno de programación independiente de la plataforma y multiproceso diseñado con el fin de crear programas y aplicaciones para Internet e Intranet.

La intención de Sun era crear un lenguaje con una estructura y una sintaxis similar a C y C++, aunque con un modelo de objetos más simple.

Los pilares en los que se sustenta Java son cinco: la programación orientada a objetos, la posibilidad de ejecutar un mismo programa en diversos sistemas operativos, la inclusión por defecto de soporte para trabajo en red, la opción de ejecutar el código en sistemas remotos de manera segura y la facilidad de uso.

En los últimos años, la empresa (que fue adquirida por Oracle) ha liberado gran parte de las tecnologías Java bajo la licencia GNU GPL.

La aplicación de Java es muy amplia. El lenguaje se utiliza en una gran variedad de dispositivos móviles, como teléfonos y pequeños electrodomésticos. Dentro del ámbito de Internet, Java permite desarrollar pequeñas aplicaciones (conocidas con el nombre de applets) que se incrustan en el código HTML de una página, para su directa ejecución desde un navegador; cabe mencionar que es necesario contar con el plug-in adecuado para su funcionamiento, pero la instalación es liviana y sencilla.

Características básicas:

- **Orientado a Objetos:** Ofrece un gran control sobre el código y una mejor organización del código al ser independientes las funciones y métodos de las veces que estas sean llamadas.
- **Muy flexible :** Java es un lenguaje especialmente preparado para la reutilización del código; permite a sus usuarios tomar un programa que hayan desarrollado tiempo atrás y actualizarlo con mucha facilidad.
- **Funciona en cualquier plataforma :** a diferencia de los programas que requieren de versiones específicas para cada sistema operativo (tales como Windows o Mac), las aplicaciones desarrolladas en Java funcionan en cualquier entorno, dado que no es el sistema quien las ejecuta, sino la máquina virtual.

- **Gratuito** : programar en Java es absolutamente gratis; no es necesario adquirir ninguna licencia, sino simplemente descargar el kit de desarrollo (**Java Development Kit** o **JDK**)
- **Lenguaje expandible**: continuando con el punto anterior, cada programador tiene la libertad de revisar y mejorar el código nativo de Java, y su trabajo puede convertirse en la solución a los problemas de muchas personas en todo el mundo.

### 2.1.5 - JavaScript:

Al igual que HTML, Javascript es un lenguaje de programación que se puede utilizar para construir sitios Web y para hacerlos más interactivos.

Aunque comparte muchas de las características y de las estructuras del lenguaje Java, fue desarrollado independientemente. El lenguaje Javascript puede interactuar con el código HTML, permitiendo a los programadores web utilizar contenido dinámico. Por ejemplo, hace fácil responder a los acontecimientos iniciados por usuarios (como introducción de datos en formularios) sin tener que utilizar CGI.

El lenguaje Javascript es opensource, por lo cualquier persona puede utilizarlo sin comprar una licencia.

### 2.1.6 - Servlets

Los Servlets son módulos que extienden los servidores orientados a petición-respuesta, como los servidores web compatibles con Java.

Por ejemplo, un servlet podría ser responsable de tomar los datos de un formulario de entrada de pedidos en HTML y aplicarle la lógica de negocios utilizada para actualizar la base de datos de pedidos de la compañía.

Los Servlets son para los servidores lo que los applets son para los navegadores. Sin embargo, al contrario que los applets, los servlets no tienen interface gráfico de usuario. Además pueden ser incluidos en muchos servidores diferentes porque el API Servlet, el que se utiliza para escribir Servlets, no asume nada sobre el entorno o protocolo del servidor. Se están utilizando ampliamente dentro de servidores HTTP

### 2.1.7 - NetBeans

NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. NetBeans IDE2 es un producto libre y gratuito sin restricciones de uso.

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs de NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos.

Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software.

### 2.1.8 - Photoshop

Adobe Photoshop es una aplicación para la creación, edición y retoque de imágenes, el cual fue y es desarrollado por la compañía Adobe Systems.

Se lanzó originalmente para computadoras Apple, pero luego saltó a la plataforma Windows.

Este programa se ha hecho muy popular, incluso fuera del ámbito informático, llegándose incluso a usar la palabra "photoshop" para hacer referencia a una foto que ha sido retocada digitalmente.

Formatos propios y/o aceptados por Photoshop:

- Los formatos propios de Photoshop son PSD y PDD, que guardan capas, canales, guías y en cualquier modo de color.
- Luego Photoshop también soporta otros formatos como PostScript, EPS, DCS, BMP, GIF, JPEG, PICT, PIFF, PNG, PDF, IFF, PCX, RAW, TGA, Scitex CT, Filmstrip, FlashPix.

## 2.1.9 - JSON

JSON es un formato de intercambio de datos de forma simple, y la cual ha sido muy popularizada como alternativa al XML en AJAX. Una de las supuestas ventajas de JSON sobre XML como formato de intercambio de datos en este contexto es que es mucho más sencillo escribir un analizador sintáctico (parser) de JSON.

En JavaScript, un texto JSON se puede analizar fácilmente usando la función `eval()`, lo cual ha sido fundamental para que JSON haya sido aceptado por parte de la comunidad de desarrolladores AJAX, debido a la ubicuidad de JavaScript en casi cualquier navegador web.

JSON se emplea habitualmente en entornos donde el tamaño del flujo de datos entre cliente y servidor es de vital importancia (de aquí su uso por Yahoo, Google, etc, que atienden a millones de usuarios).

Está basado en un subconjunto del Lenguaje de Programación JavaScript, Standard ECMA-262 3rd Edition - Diciembre 1999. JSON es un formato de texto que es completamente independiente del lenguaje pero utiliza convenciones que son ampliamente conocidos por los programadores de la familia de lenguajes C, incluyendo C, C++, C#, Java, JavaScript, Perl, Python, y muchos otros.

Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos.

JSON está constituido por dos estructuras:

- Una colección de pares de nombre/valor. En varios lenguajes esto es conocido como un objeto, registro, estructura, diccionario, tabla hash, lista de claves o un arreglo asociativo.
- Una lista ordenada de valores. En la mayoría de los lenguajes, esto se implementa como arreglos, vectores, listas o secuencias.

### 2.1.10 - Demonio

Un demonio, daemon o dæmon (de sus siglas en inglés Disk And Execution Monitor), es un tipo especial de proceso informático no interactivo, es decir, que se ejecuta en segundo plano en vez de ser controlado directamente por el usuario. Este tipo de programas se ejecutan de forma continua (infinita), vale decir, que aunque se intente cerrar o matar el proceso, este continuará en ejecución o se reiniciará automáticamente. Todo esto sin intervención de terceros y sin dependencia de consola alguna.

Se trata de una aplicación UNIX que está alerta permanentemente en un servidor Internet para realizar determinadas tareas como, por ejemplo, enviar un mensaje de correo electrónico o servir una página web, gestión una alarma....



## 2.2 - Herramientas de Almacenamiento

### 2.2.1 - Bases de datos Mysql

MySQL Database Server es la base de datos de código fuente abierto más usada del mundo. Su ingeniosa arquitectura lo hace extremadamente rápido y fácil de personalizar. La exclusiva separación del core server del manejador de tablas, permite funcionar a MySQL bajo control estricto de transacciones o con acceso a disco no transaccional ultrarrápido.

MySQL es un sistema de administración relacional de bases de datos. Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido. MySQL es software de fuente abierta. Cualquier persona puede bajar el código fuente de MySQL y usarlo sin pagar. Cualquier interesado puede estudiar el código fuente y ajustarlo a sus necesidades.

MySQL usa el GPL (GNU General Public License) para definir qué puede hacer y que no puede hacer con el software en diferentes. MySQL Server fue desarrollado inicialmente para manejar grandes bases de datos mucho más rápidamente que las soluciones existentes y ha sido usado exitosamente por muchos años en ambientes de producción de alta demanda. A través de constante desarrollo, MySQL Server ofrece hoy una rica variedad de funciones. Su conectividad, velocidad y seguridad hacen a MySQL altamente satisfactorio para el uso de bases de datos en Internet.

## 2.3 - Herramientas de Hospedaje / Servidor

### 2.3.1 - Servidor web:

Un servidor web es un programa que implementa el protocolo HTTP (hypertext transfer protocol). Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas web o páginas HTML (hypertext markup language), textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos.

Un servidor web se encarga de mantenerse a la espera de peticiones HTTP llevada a cabo por un cliente HTTP que solemos conocer como navegador. El navegador realiza una petición al servidor y éste le responde con el contenido que el cliente solicita.

Sobre el servicio web clásico podemos disponer de aplicaciones web. Éstas son fragmentos de código que se ejecutan cuando se realizan ciertas peticiones o respuestas HTTP.

Hay que distinguir entre:

#### Aplicaciones en el lado del cliente:

El cliente web es el encargado de ejecutarlas en la máquina del usuario. Son las aplicaciones tipo Java o javascript: el servidor proporciona el código de las aplicaciones al cliente y éste, mediante el navegador, las ejecuta. Es necesario, por tanto, que el cliente disponga de un navegador con capacidad para ejecutar aplicaciones (también llamadas scripts).

Normalmente, los navegadores permiten ejecutar aplicaciones escritas en lenguaje javascript y java, aunque pueden añadirse mas lenguajes mediante el uso de plugins.

#### Aplicaciones en el lado del servidor:

El servidor web ejecuta la aplicación, la cual una vez ejecutada, genera código HTML y el servidor toma este código para enviarlo al cliente por medio del protocolo HTTP. Las aplicaciones de servidor suelen ser la opción por la que se opta en la mayoría de las ocasiones para realizar aplicaciones web.

La razón es que, al ejecutarse ésta en el servidor y no en la máquina del cliente, éste no necesita ninguna capacidad adicional, como sí ocurre en el caso de querer ejecutar aplicaciones javascript o java. Así pues, cualquier cliente dotado de un navegador web básico puede utilizar este tipo de aplicaciones.

### 2.3.2 - Cloud Computing:

El *cloud computing* es un modelo tecnológico que permite el acceso adaptado y bajo demanda en red a un conjunto compartido de recursos de computación configurables compartidos (por ejemplo: redes, servidores, equipos de almacenamiento, aplicaciones y servicios), que pueden ser rápidamente aprovisionados y liberados con un esfuerzo de gestión reducido o interacción mínima con el proveedor del servicio, es decir, se basa en la posibilidad de ofrecer servicios de computación a través de internet.

La computación en la nube son servidores desde Internet encargados de atender las peticiones en cualquier momento. Se puede tener acceso a su información o servicio, mediante una conexión a internet desde cualquier dispositivo móvil o fijo ubicado en cualquier lugar. Sirven a sus usuarios desde varios proveedores de alojamiento repartidos frecuentemente por todo el mundo. Esta medida reduce los costes, garantiza un mejor tiempo de actividad y que los sitios web sean invulnerables a los hackers, a los gobiernos locales y a sus redadas policiales.

Los modelos de soluciones en la nube eliminan de esta manera la necesidad de grandes inversiones y costes fijos, al no tener la necesidad de tener las aplicaciones instaladas en el ordenador de cada usuario , pudiendo ser accedida desde cualquier lugar. Los proveedores en empresas de servicios que ofrecen de forma flexible e instantánea la capacidad de computación bajo demanda.

### 2.3.2.1 - AWS

Amazon Web Services (AWS) es una colección de servicios de programación en la nube (también llamados servicios web) que componen una plataforma cloud computing, que son ofrecidos a través de Internet por Amazon.com. Los más conocidos de estos servicios son Amazon EC2 y Amazon S3.

El servicio proporciona una capacidad de computación de gran tamaño (muchos servidores interactuando en paralelo), lo que hace que sea mucho más rápido y más barato que la adquisición y mantenimiento de servidores físicos.

Dentro de AWS se nos ofrece tanto la posibilidad del hospedaje, como la instalación dentro del mismo de distintas herramientas de programación y desarrollo que permitan la ejecución de todos nuestros programas.

### 2.3.3 - Apache Tomcat:

Apache Tomcat (o Jakarta Tomcat) es un software desarrollado con Java (con lo cual puede funcionar en cualquier sistema operativo, con su máquina virtual java correspondiente) que sirve como servidor web con soporte de servlets y JSPs.

Tomcat es mantenido y desarrollado por miembros de la Apache Software Foundation y voluntarios independientes. Los usuarios disponen de libre acceso a su código fuente y a su forma binaria en los términos establecidos en la Apache Software License.

Tomcat es un servidor web con soporte de servlets y JSPs. Tomcat no es un servidor de aplicaciones. Incluye el compilador Jasper, que compila JSPs convirtiéndolas en servlets. El motor de servlets de Tomcat a menudo se presenta en combinación con el servidor web Apache.

Tomcat puede funcionar como servidor web por sí mismo. En sus inicios existió la percepción de que el uso de Tomcat de forma autónoma era sólo recomendable para entornos de desarrollo y entornos con requisitos mínimos de velocidad y gestión de transacciones. Hoy en día ya no existe esa percepción y Tomcat es usado como servidor web autónomo en entornos con alto nivel de tráfico y alta disponibilidad.

## 2.4 - Arquitectura

### 2.4.1 - Cliente-servidor

Normalmente en Internet se usa la arquitectura cliente-servidor, la cual se basa en que entre todos los ordenadores que están en la red, unos ofrecen servicios (los llamados servidores) y otros usan esos servicios (los denominados clientes). Una petición de una página WEB es una petición de un ordenador (cliente) al que hospeda estas páginas (servidor).

### 2.4.2 - Web

Consiste en el uso de un protocolo HTTP para que desde un cliente se solicite un documento dentro de la red , y un servidor HTTP le sirva una página en formato HTML.

La arquitectura web está basada en el uso de clientes ligeros y servidores distribuidos con diferentes funciones.

## 2.5 - Conclusiones al estado del arte.

Una vez definidos los conceptos básicos que han de ser tenidos en cuenta para el completo entendimiento del proyecto, se es capaz de detallar y comprender el porque de las tecnologías seleccionadas para este proyecto.

Tal y como se detallaba dentro del punto 1.5 de este mismo documento, una de las bases por las que se ha de partir en este proyecto, es el coste de obtención y mantenimiento de los recursos necesarios para su funcionamiento. Estamos hablando de que esta solución se implementará dentro de una empresa de un tamaño más bien reducido, por lo que esta no tiene la capacidad de pagar grandes costes por mantenimientos y licencias.

De esta manera, se ha seleccionado un alojamiento vía Amazon en la cual la empresa no requiere la compra de un servidor, mantenimiento, sistemas operativos,, etc, además de que el rendimiento será mayor que mediante un servidor propio, y con un coste temporal bastante más reducido.

Con respecto al mantenimiento de los datos, se ha optado por Mysql por varios motivos, entre los que en primer lugar se encuentra el precio, dado que a volúmenes no muy elevados, es uno de los gestores más económicos del mercado. Por otro lado, se encuentra la buena adaptación que esta tiene con los servidores amazon, estando esta bbdd directamente incluida dentro de los servicios de esta compañía.

A la hora de realizar la programación, se ha optado por la utilización de una tecnología html, jsp, java, mediante netbeans, en primer lugar por la necesidad de que la aplicación vaya a nivel web, y java por la gran integración cliente servidor que esta ofrece, haciendo que esta aplicación pueda crecer mucho a lo largo de los años.

Mediante estas opciones, podremos tener una aplicación segura, fiable, fácilmente mantenible, y con posibilidades de futuras mejoras, mediante un coste reducido, lo que a buen seguro hace que el cliente final quede completamente satisfecho con el resultado obtenido.

## 3 - Estudio Previo y Definición de Requisitos

### 3.1 - Introducción

A fecha 8 de Junio de 2013, la empresa "Vellés System Consulting S.L.", se puso en contacto con P.R.R.B.S, solicitando la prestación de servicios de cara a la realización de un amplio estudio sobre las metodologías de trabajo y comunicación en su ámbito interno, con el fin de poder darle solución y de esta manera evitar problemas como los que venía cursando hasta el momento.

El objetivo principal del estudio previo se basa en el entendimiento real de lo que el cliente necesita, y quiere tener, hasta conseguir definir unos requisitos claros, que puedan ser utilizados como base para el resto de fases del proyecto.

Tras una serie de reuniones, la cuales comenzaron con un grado de incertidumbre muy alto, y a muy alto nivel de conocimiento del detalle que se podría requerir, se fue bajando poco a poco el nivel y se fue entendiendo cual era la situación actual de la compañía, y cuál era la necesidad del cliente.

## 3.2 - Fase de Estudio Previo

### 3.2.1 - Situación Actual

Tras los primeros contactos con la empresa V.S.C., se ha podido ver cuál es la situación actual de la empresa. Esta se encuentra dividida en diversos departamentos, y secciones, por las cuales va pasando la información de cada uno de los proyectos en cada uno de los ciclos de vida del mismo, pero actualmente no existe ninguna herramienta de comunicación interna que permita el acceso a la información del resto de departamentos.

Por otro lado, se ha detectado un problema a la hora de la elección de los recursos, dado que al no encontrarse la empresa comunicada los responsables de la planificación y organización de los proyectos, desconocen los distintos perfiles, y aptitudes con los que cuenta dentro de la empresa, lo que hace que los proyectos no sean realizados por los recursos más adecuados para ello, ni por los más disponibles, existiendo de esta manera recursos muy sobrecargados de trabajo, y recursos muy liberados.



### 3.2.2 - Descripción de la Solución

Tras realizar el estudio de las necesidades del cliente, y las posibles mejoras que podría llevar a cabo, por parte de P.R.R.B.S.. se propone la creación de un aplicación con acceso desde entorno web por cualquier miembro de la empresa, dentro de la cual se pueda:

- Gestionar proyectos --> Planificación, asignación, espacio en disco donde pueda almacenarse la información relativa al proyecto, y pueda ser accedida en cualquier momento, etc
- Posibilidad de asignación automática de recursos --> Se propone la inclusión de un motor de asignación automática, que proponga por cada uno de los proyectos planificados ( teniendo en cuenta la fecha planificada, perfil del proyecto, perfil de los empleados, disponibilidad, etc), el recurso más recomendable para llevar a cabo dicha tarea.
- Gestión interna de acciones propias de los empleados --> Como puede ser la imputación de horas diaria, solicitud de vacaciones, modificación de datos propios, etc
- Sistema de alarmado --> Mediante este sistema se avisará a cada uno de los empleados de cuando comienza, finaliza cada tarea que tiene asignada. Por otro lado, se establecerán unos límites de sensibilidad, a partir de los cuales si un proyecto lo supera, serán avisados los responsables para que tomen las medidas que se consideren más oportunas.
- Reporting interno de la empresa --> Se proporcionarán herramientas, mediante las cuales puedan realizarse tareas de reporting interno de la empresa, y de esta forma poder conocer el estado actual y futuro de la misma.

Mediante al solución a alto nivel que ha realizado V.S.C., el usuario final define unos requisitos propios que definirán la solución final a ser implementada.

### 3.3 - Requisitos de Usuario

Finalmente y una vez finalizado el estudio previo de las necesidades de nuestro cliente, se han definido los requisitos de usuario que establecen el alcance final del proyecto

Existen dos tipos de requisitos, tan importantes los unos como los otros, dado que si no se cumplen ambos, el proyecto podría no cumplir los criterios mínimos de aceptación:

- Requisitos Funcionales --> Se trata de una característica requerida dentro del sistema, la cual conlleva una codificación, y una nueva funcionalidad en la aplicación, como puede ser la inclusión en una aplicación de un nuevo botón para la generación de un nuevo informe.
- Requisitos No Funcionales --> Se trata de una característica técnica que se le imponen a un proyecto, y que no implican nueva funcionalidad, como pudiera ser la obligatoriedad de utilizar un lenguaje de programación determinado, la prohibición de ocupar más de cierto espacio en BBDD, etc.

De esta manera, se definirán los requisitos mediante los siguientes campos:

- Requisito:

Identificador único de requisito, que servirá para ir trazando en las siguientes fases del proyecto, el avance del mismo

Se definirá como XX\_??? , donde XX podrá ser:

RF --> Requisito Funcional

RNF --> Requisito No Funcional

- Descripción del Requisito:

Título explicativo del requisito y descripción detallada del mismo.

- Prioridad:

- Esencial(E) --> Requisito absolutamente necesario para la implementación del proyecto.

- Opcional (O) --> Requisito NO absolutamente necesario para la implementación del proyecto.

- Departamento:

- Departamento para el que va dirigido el requisito

- Criterios de Aceptación:

- Criterios de aceptación y métricas asociadas

Finalmente los requisitos identificados para este proyecto serán los siguientes:

Tabla 1 - Requisitos Usuario

Requisito	Descripción del Requisito	Prioridad	Departamento	Criterios de Aceptación	Resultado
<b>Requisitos Funcionales</b>					
RF_001	La aplicación tendrá una página de autenticación, la cual evite el acceso de personas no autorizadas	E	Seguridad	Acceso permitido únicamente a personas previamente dadas de alta en el sistema.	OK
RF_002	Las distintas funcionalidades de la aplicación estarán limitadas, dependiendo del perfil del usuario que lo utilice. Dentro de la aplicación se definirán 7 tipos de perfiles de acceso a los datos: <ul style="list-style-type: none"> <li>• Ejecutivo Comercial</li> <li>• Jefe de Proyecto</li> <li>• Analista</li> <li>• Programador</li> <li>• Recursos Humanos</li> <li>• Administrador</li> <li>• Otros</li> </ul>	E	Seguridad	Acceso restringido a las distintas páginas, y permitido, según perfil.	OK
RF_003	Únicamente el administrador de la aplicación podrá modificar los derechos de cada uno de los perfiles dentro de la aplicación.	E	Seguridad	Derechos de la aplicación para el campo derechos, únicamente disponible para el administrador.	OK
RF_004	La aplicación contendrá una página inicial, desde la cual se tenga acceso a toda la información de : <ul style="list-style-type: none"> <li>• Cartera</li> <li>• Empleados</li> <li>• Alarmado</li> <li>• Informes</li> <li>• Administración</li> </ul>	E	Todos	Página inicial con redirección a las distintas funcionalidades.	OK
RF_005	Se dotará a la aplicación de un sistema de	E	Todos	Sistema de mensajería	OK

	mensajería personalizado por cada usuario, y con acceso desde cualquier punto de la aplicación			creado.	
RF_006	Se permitirá dar de alta nuevos proyectos dentro de la aplicación, tanto en información, como en planificación	E	Gestión de Proyectos	Un usuario puede dar de alta un proyecto, y planificarlo correctamente.	OK
RF_007	El sistema dará la opción de realizar una planificación manual o automática de los recursos. Se han de incluir las operaciones de cálculo que permitan obtener el recurso más adecuado para dicho proyecto en el momento de la búsqueda, de forma automática.	E	Gestión de Proyectos	Al tratar de planificar un proyecto, se incluye la opción de poder planificarlo tanto manualmente como automáticamente.	OK
RF_008	Se permitirá modificar cualquier información de un proyecto ya existente dentro de la Base de Datos	E	Gestión de Proyectos	La información de los proyectos es modificable en todos sus valores.	OK
RF_009	Se permitirá eliminar cualquier proyecto ya existente dentro de la Base de Datos	E	Gestión de Proyectos	Se puede realizar la eliminación de los datos de un proyecto.	OK
RF_010	Se incluirá un buscador de proyectos, dentro de la aplicación	E	Gestión de Proyectos	Los proyectos se pueden buscar por código y nombre.	OK
RF_011	Cualquier empleado, podrá ver los datos que de él tiene almacenado la empresa desde el área de empleados, así como modificar su clave de acceso a la aplicación.	E	R.R.H.H.	Los empleados pueden acceder desde la aplicación a sus datos personales.	OK
RF_012	Cualquier empleado, podrá ver los productos/proyectos que actualmente tiene asignado, así como su estado, y fase actual.	E	Gestión de Proyectos	Los empleados pueden acceder desde la aplicación a sus proyectos asignados.	OK
RF_013	Desde el área de empleados, se han de	E	Gestión de Proyectos	Los empleados pueden realizar	OK

	poder realizar la imputaciones diarias de trabajo realizado, así como ver las imputaciones realizadas.			imputaciones desde la aplicación.	
RF_014	Desde el área de empleados, se incluirá un acceso a la zona de solicitudes, en las cuales, cualquier empleado podrá generar una solicitud ( Vacaciones, Ausencias, etc), y los respectivos aprobadores, tendrán la capacidad de aprobarlas o rechazarlas.	E	R.R.H.H. Gestión de Proyectos	Los empleados pueden realizar solicitudes a sus responsables, y estos aprobarlas , rechazarlas.	OK
RF_015	Se incluirá desde la pagina principal un enlace al sistema de gestión de alarmado	E	Gestión de Proyectos	Desde la pagina principal, se puede acceder a la gestión de las alarmas.	OK
RF_016	Se permitirá dar de alta nuevas alarmas dentro de la aplicación, para ciertos perfiles	E	Gestión de Proyectos	Se permite la creación de nuevas alarmas en el sistema.	OK
RF_017	Se permitirá modificar cualquier información de una alarma ya existente dentro de la Base de Datos	E	Gestión de Proyectos	La alarmas pueden modificarse	OK
RF_018	Se permitirá eliminar cualquier alarmas ya existente dentro de la Base de Datos	E	Gestión de Proyectos	Las alarmas pueden eliminarse del sistema.	OK
RF_019	Se incluirá un buscador de alarmas, dentro de la aplicación	E	Gestión de Proyectos	Las alarmas pueden ser buscadas por nombre de alarma.	OK
RF_021	Se permitirá crear nuevos informes, mediante la selección de los datos de las tablas que actualmente existen en Base de datos., así como ejecutarlos, ver los resultados, y se ofrecerá la posibilidad de almacenarlos como	E	Gestión de Proyectos	Existe la capacidad de crear informes, exportarlos a Excel y guardarlos como favoritos.	OK

	favoritos, para no tener que volver a repetir el informe.				
RF_022	Se dará la posibilidad de recuperar los informes guardados anteriormente como favoritos	E	Gestión de Proyectos	Los informes pueden volverse a ejecutar en un futuro	OK
RF_023	Se dará la posibilidad de planificar informes guardados anteriormente como favoritos para que se ejecuten a una hora determinada	E	Gestión de Proyectos	Se pueden planificar la ejecución de los informes para que se ejecuten automáticamente.	OK
RF_024	Se dará la posibilidad de ver la planificación actual de los informes guardados anteriormente como favoritos. Todos los usuarios tendrán acceso a los informes del resto de usuarios, pero solo el usuario creador, podrá modificar, borrar, o cambiar la planificación de un informe.	E	Gestión de Proyectos	Se puede ver el listado de informes planificados, así como modificar los datos de esta planificación.	OK
RF_025	Se solicita que puedan darse de alta los empleados de la empresa, por parte de Recursos Humanos, con la posibilidad de incluir todos los conocimientos que este empleado posee.	E	R.R.H.H.	Puede gestionarse la cartera de alta de empleados correctamente.	OK
RF_026	Se solicita que puedan modificarse los datos de los empleados de la empresa, por parte de Recursos Humanos.	E	R.R.H.H.	Puede modificarse la cartera de empleados correctamente.	OK
RF_027	Se solicita que puedan darse de baja los datos de los empleados de la empresa, por parte de Recursos Humanos.	E	R.R.H.H.	Puede gestionarse la cartera de baja de empleados correctamente.	OK
<b>Requisitos No Funcionales</b>					
RNF_001	La información será almacenada dentro de una Base de datos de tipo MySQL	O	L.A.N.	Los datos se almacenan en Mysql	OK

RNF_002	El sistema deberá ser capaz de funcionar bajo un entorno Windows	E	L.A.N.	El sistema funciona bajo entorno Windows	<b>OK</b>
RNF_003	El interface de la aplicación debe guardar semejanza y coherencia en todas las pantallas de la aplicación	O	L.A.N.	Todas las pantallas de la aplicación siguen el mismo estándar.	<b>OK</b>
RNF_004	La información almacenada en el sistema nunca debe ser vista sin el software desarrollado	E	L.A.N. Seguridad	Solo puede accederse a la información de BBDD mediante la aplicación desarrollada.	<b>OK</b>
RNF_005	La solución aportada, podrá ser accedida mediante "Internet Explorer" dentro de los ordenadores de la compañía	E	L.A.N.	La aplicación funciona correctamente bajo Windows Explorer.	<b>OK</b>
RNF_006	Los campos de tipo fecha, serán insertados mediante un desplegable de tipo calendario	O	L.A.N.	Todas las inserciones son realizadas mediante desplegables.	<b>OK</b>
RNF_007	El proyecto tendrá que ser llevado a cabo con una duración máxima no superior a un año natural	E	Gestión de Proyectos	La duración del proyecto es inferior a un año	<b>OK</b>
RNF_008	La aplicación será mejorada para su visualización con resolución 1366x768	O	L.A.N.	Bajo esta resolución , la aplicación se ve correctamente.	<b>OK</b>



## 4 - Estudio Analítico del Proyecto

### 4.1 - Ciclo de vida

Llegados a este punto del proyecto, es necesario establecer cuál va a ser el ciclo de vida del mismo, y el cual nos mostrará cual va a ser la metodología y fases de las que constará el proyecto, las cuales será necesario ir completando secuencialmente.

La correcta definición de este, nos permitirá facilitar el control de tiempos, recursos de todo tipo ( personal, equipos, etc), fechas, entregables, etc.

Dentro de los distintos modelos de ciclo de vida ( lineal, espiral, prototipado, etc), se ha considerado que el más conveniente para llevar a cabo de forma correcta este proyecto en concreto es un tipo de ciclo de vida lineal o en cascada, el cual consiste en descomponer la actividad global del proyecto en fases que se suceden de manera lineal, dividiendo las tareas entre equipos y estableciendo los tiempos de actuación de cada uno.

Los motivos por los que se ha escogido este tipo de ciclo de vida, se basa en:

- \* Los requisitos están completamente definidos y se conoce al completo el alcance que se quiere que tenga la aplicación, por lo que no se considera la necesidad de ir realizando prototipos, o entregas parciales, para verificar si ciertamente es lo que se requiere.
- \* Es el más comúnmente utilizado dentro de la empresa, por lo que todos los empleados están acostumbrados a su aplicación y desarrollo.
- \* No es necesario información de fases posteriores para poder dar por finalizada ninguna fase anterior, por lo que cada fase puede cerrarse por completo, con anterioridad al comienzo de la siguiente.

Este modelo constará de las siguientes fases:

- Definición de necesidades y requisitos ( Ya finalizada)
- Análisis
- Diseño
- Codificación
- Pruebas
- Despliegue
- Mantenimiento y operación

Las cuales estarán ligadas, tal y como se muestra en la siguiente figura:

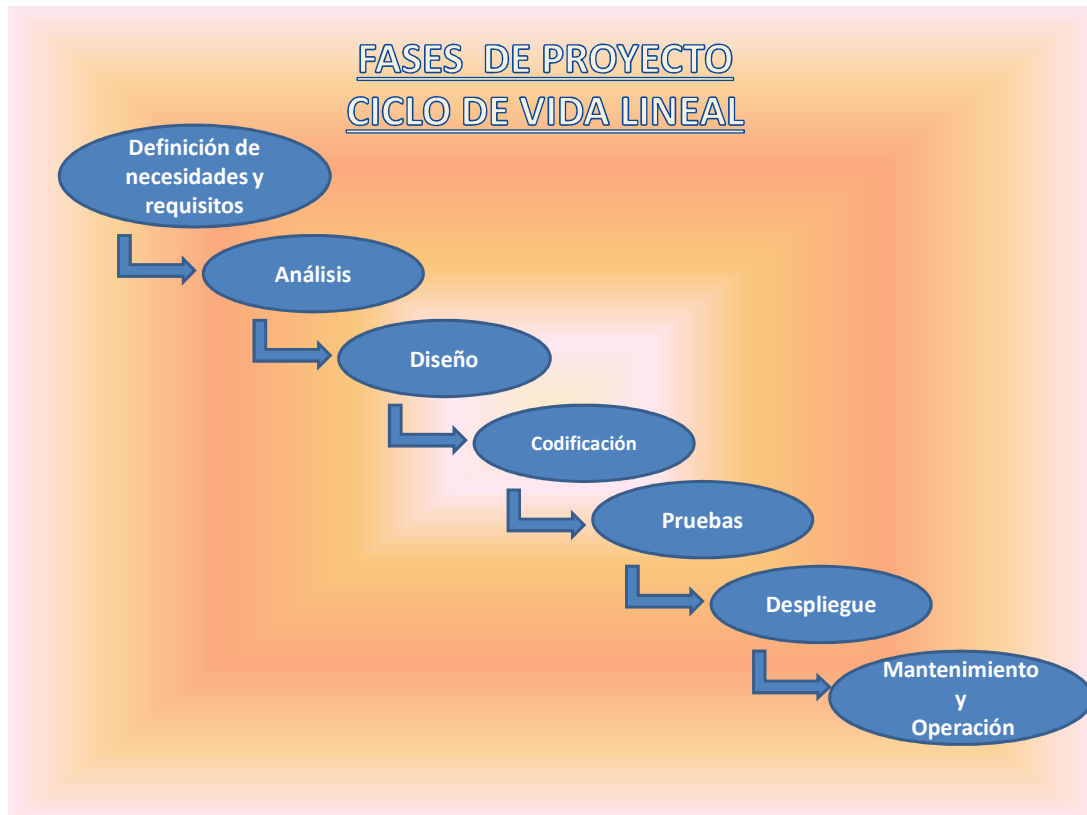


Figura 1. - Ciclo de Vida de un proyecto

#### Definición de necesidades y requisitos

Se basa en lograr comprender la problemática existente, y la solución que el usuario espera de nuestra empresa

#### Análisis

La salida del análisis debe incluir una especificación funcional y un análisis estructurado que contiene los requerimientos para el nuevo sistema, los cuales el usuario debe leer, analizar y señalar lo que él quiere. Es decir, ha de incluir la solución que la empresa dará a cada uno de los requisitos de forma funcional, de manera que el usuario final pueda comprender lo que se le ofrece. Busca el "que" debe hacer el sistema, pero no como.

#### Diseño

Se basa en la descripción de la solución propuesta dentro del análisis a un nivel tecnológico y estructurando correctamente los cambios y componentes que serán necesarios para poder llevar a cabo la solución. Busca el "como" se puede implementar la solución requerida.

### Codificación

Es la fase de programación o escritura del código. Lo que se produce en el diseño se lleva a código.

### Pruebas

En esta etapa se planea el diseño de casos de prueba con el fin de "asegurar" el correcto funcionamiento del sistema. así como estas son ejecutas.

### Despliegue

Se basa en la puesta en productivo de la solución propuesta, comenzando desde dicho momento el usuario a poder utilizar la nueva aplicación.

### Mantenimiento y Operación

En esta etapa se hace el manteamiento correspondiente al sistema y además mejorar algunos procesos para su correcto funcionamiento.

## 4.2 - Organigrama y Recursos Humanos

De cara a conocer el presupuesto por el que se ofertará el proyecto al completo, definir las fases, etc. es necesario establecer el RBS (Resource Breakdown Structure) el cual se basa en un árbol jerárquico que representa los recursos humanos y materiales de los que constará el proyecto.

Los principales objetivos del RBS se basan en :

- \* Mostrar de forma grafica la organización humana del proyecto
- \* Maximizar el uso de los conocimientos del personal disponible
- \* Reflejar la estructura de recursos materiales de los que constará el proyecto, así como sus coste asociados. ( esto será tratado dentro del punto 10 de este mismo proyecto)

## 4.2.1 - Recursos Humanos

Se hace necesario a estas alturas del proyecto, el establecimiento de un *Plan de Recursos Humanos*, mediante el cual identificar y documentar los roles y responsabilidades que serán requeridas para llevar a cabo el proyecto.

Para este proyecto en particular, se han considerado necesarios los siguiente puestos:

### - Jefe de Proyecto

Es el responsable del proyecto, así como el interlocutor con el cliente, y con la dirección de la propia compañía. Entre sus tareas, se encuentra la definición de los plazos, costes, aplicación del plan de desarrollo y plan de calidad , así como participación activa en las fases de Análisis, diseño e Integración, proporcionando su validación.

### - Analista Funcional

Es el encargado del estudio de los requisitos impuestos por el usuario realizando una labor de asesoramiento y capacitación, de cara a evaluar la viabilidad técnica de los desarrollos de las aplicaciones que se han de ejecutar, y preparar y elaborar toda la documentación técnica y de usuario de cada aplicación, asegurando una correcta explotación y su óptimo rendimiento.

### - Diseñador

Es el encargado de la toma del análisis realizado por el analista funcional, y realizar una análisis a más bajo nivel que defina de que manera va a ser implementada la solución propuesta dentro del propio sistema.

### - Programador

Es el que escribe, depura y mantiene el código fuente de un programa informático, es decir, del conjunto de instrucciones que ejecuta el hardware de una computadora para realizar una tarea determinada.

### - Responsable de Calidad

Es el responsable de realizar las revisiones periódicas que aseguren que se establecen, implementan y mantienen los procesos necesarios para el sistema de calidad.

De esta manera, tendremos un sistema jerárquico como el que se muestra a continuación:



Figura 2. - Organigrama de Recursos Humanos

Como se puede observar, se trata de un equipo de trabajo de tipo Centralizado Controlado, dentro del cual el Jefe de Proyecto, se erige como el responsable máximo, y el resto del equipo colabora en la solución a aportar.

#### Dimensionamiento del Equipo de Trabajo

Dada la estimación de tiempo y coste que va a ser presentada al cliente de cara a la compra del proyecto, el dimensionamiento del equipo de trabajo será como se muestra a continuación:

Recurso	Número de Recursos
Jefe de Proyecto	1
Analista Funcional	1
Diseñador	1
Programadores	2
Responsable de Calidad	1

Tabla 2 - Tabla de recursos para el proyecto

Por lo tanto, de cara a la realización del proyecto, necesitaremos de la participación de 6 recursos, aunque no todos ellos a tiempo completo.

Una vez determinada la planificación, podremos determinar en qué fases del proyecto participará cada recurso, y con qué esfuerzo.

### 4.2.2 - Planificación

De cara al éxito final del proyecto, es de vital importancia establecer una correcta planificación del mismo, la cual dependerá de factores como:

- Tiempo de ejecución esperado
- Recursos Disponibles
- Precio global estimado para el proyecto

Para la planificación se decidió realizar un diagrama de Gantt para comprobar la validez del proyecto, y mediante el cual se puede representar el desarrollo de las distintas actividades de las que consta, de forma fácil y rápida. De esta manera, podremos observar en una línea del tiempo las fases desarrolladas para la consecución del proyecto que se está llevando a cabo.

El usuario ha marcado como requisito básico que el producto esté listo y en productivo en un plazo máximo de un año natural desde el comienzo del mismos, por lo que estableceremos la duración y planificación basándonos en dicha premisa.

Mediante la siguiente grafica podremos ver la duración, inicio, fin y secuencialidad de cada una de las fases que serán requeridas para llevar el proyecto a cabo:

	Task Name	Duration	Start	Finish	Predecessors
1	Estudio Previo y Generación de Requisitos	30 days	Mon 02/09/13	Fri 11/10/13	
2	Análisis de Proyecto (Macrodiseno)	60 days	Mon 14/10/13	Fri 03/01/14	1
3	Diseño de Proyecto (Microdiseno)	50 days	Wed 15/01/14	Tue 25/03/14	2FS+7 days
4	Construcción	60 days	Wed 26/03/14	Tue 17/06/14	3
5	Pruebas	20 days	Wed 18/06/14	Tue 15/07/14	4
6	Despliegue	7 days	Wed 16/07/14	Thu 24/07/14	5
7	Mantenimiento	30 days	Fri 25/07/14	Thu 04/09/14	6

Figura 3 - Planificación general del proyecto

El diagrama de Gantt relativo a este detalle, es como indica la figura que se muestra a continuación:

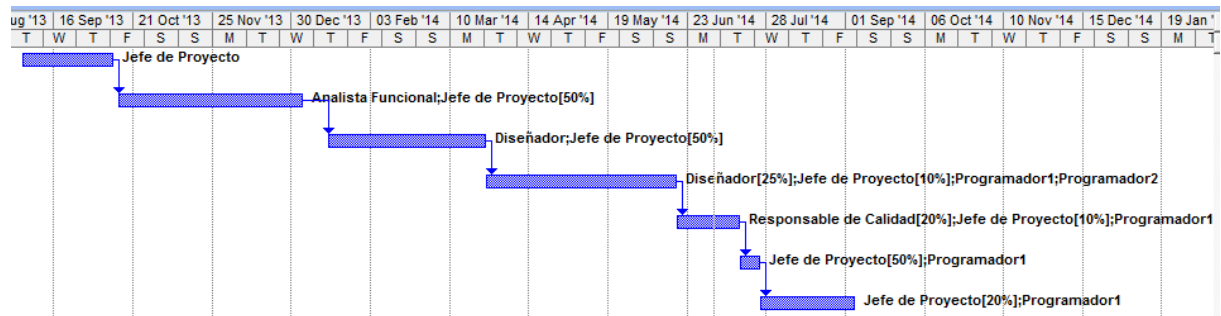


Figura 4 - Diagrama de Gantt del proyecto

Dentro de esta figura podemos observar las fecha inicio fin de cada fase, así como los recursos que a los que afecta cada una de las fases.

En la siguiente figura, se muestran los porcentajes de tiempo asignados a cada recurso en cada fase del proyecto

Resource Name	Assignment Units
<input type="checkbox"/> Analista Funcional	
<i>Análisis de Proyecto (Macrodiseno)</i>	100%
<input type="checkbox"/> Diseñador	
<i>Diseño de Proyecto (Microdiseno)</i>	100%
<i>Construcción</i>	25%
<input type="checkbox"/> Responsable de Calidad	
<i>Pruebas</i>	20%
<input type="checkbox"/> Jefe de Proyecto	
<i>Despliegue</i>	50%
<i>Análisis de Proyecto (Macrodiseno)</i>	50%
<i>Diseño de Proyecto (Microdiseno)</i>	50%
<i>Construcción</i>	10%
<i>Mantenimiento</i>	20%
<i>Pruebas</i>	10%
<i>Estudio Previo y Generación de Requisitos</i>	100%
<input type="checkbox"/> Programador1	
<i>Construcción</i>	100%
<i>Pruebas</i>	100%
<i>Despliegue</i>	100%
<i>Mantenimiento</i>	100%
<input type="checkbox"/> Programador2	
<i>Construcción</i>	100%

Figura 5 - Porcentajes de tiempo por fase



Por lo tanto los recursos tendrán que ir incorporándose progresivamente al proyecto, con unas fechas de asignación total o parcial como las que siguen:

	Fecha Inicio Participación	Fecha Fin Participación
Jefe de Proyecto	02/09/2013	04/09/2014
Analista Funcional	14/10/2013	03/01/2014
Diseñador	15/01/2014	17/06/2014
Responsable de Calidad	18/06/2014	15/07/2014
Programador 1	26/03/2014	04/09/2014
Programador 2	26/03/2014	17/06/2014

Tabla 3 - Planificación general del proyecto por perfil

## 5 - Diseño de la Aplicación

### 5.1 - Introducción

Para la realización del diseño de la aplicación y contemplando el análisis realizado, ha sido necesario el realizar diferentes modelados y diagramas, con el fin de verificar y delimitar la solución final propuesta.

El diseño correspondiente al análisis realizado se ha subdividido en varias partes, cada una de ellas, partes fundamentales para la comprensión de la solución final

Las partes que se mencionan en el párrafo anterior son las siguientes:

- ✓ Casos de uso
- ✓ Modelo de datos
- ✓ Modelo entidad/relación de la BBDD
- ✓ Modelo relacional de la BBDD
- ✓ Esquema de BBDD
- ✓ Diseño general del diagrama de clases
- ✓ Diseño del proceso de ejecución de alertas (procesos Demonios)
- ✓ Diagramas de secuencia

Con el análisis pertinente y el diseño de la aplicación, realizando todos estos pasos se puede llegar a una conclusión pertinente respecto del desarrollo definitivo de la aplicación de gestión de cartera

## 5.2 - Casos de Uso

### 5.2.1 - Introducción

Los casos de uso se trata de una descripción de los pasos o actividades que deberían llevarse a cabo para realizar una tarea, así como las interacciones entre actores y sistema.

Los casos de uso que se expondrán a continuación será la primera acción que se va a realizar como parte del diseño de la aplicación que se va a desarrollar.

Los casos de uso ayudarán a definir y describir lo que el sistema debe de hacer, y como debe interactuar con los usuarios que utilizarán el sistema.

En este punto se va desarrollar primero el diagrama de Casos de Uso y a continuación y como parte más importante de este punto, la especificación o descripción de los casos de uso que realmente aclarará la forma de interacción entre usuario y sistema.

De cada caso de uso se mostrará en forma de tabla la siguiente información:

- ❖ Caso de Uso: Nombre descriptivo del caso de uso
- ❖ Descripción: objetivo del caso de uso
- ❖ Actores: entidades que efectúan acciones en cada caso de uso.
- ❖ Precondiciones: factores necesarios para la realización del caso de uso.
- ❖ Postcondiciones: factores que se darán una vez finalizado el caso de uso.
- ❖ Flujo Normal: pasos habituales que se seguirán durante la ejecución del caso de uso.
- ❖ Flujo Alternativo: caminos opcionales que podrían darse durante la ejecución del caso de uso.

## 5.2.2 - Diagrama de los Casos de Uso

Mediante el diagrama de casos de uso, podemos observar el comportamiento que tiene el sistema desde el punto de vista del usuario, atendiendo a la funcionalidad del mismo, y dejando a un lado la implementación que este tiene a un bajo nivel.

Este tipo de diagramas es muy útil debido a su simplicidad, y a la facilidad de interpretación visual del mismo.

A continuación se presenta el diagrama de casos de uso perteneciente a la aplicación de gestión de cartera:



Figura 6 - Diagrama de casos de uso de la aplicación

## 5.2.3 - Especificación de los Casos de Uso

### 5.2.3.1 - General

<b>Caso de Uso</b>	Logado en el Sistema
<b>Descripción</b>	Logado dentro de la aplicación
<b>Actores</b>	Usuario
<b>Precondiciones</b>	
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>▪ Usuario logado en el sistema</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Se visualiza una pantalla donde introducir el usuario y login</li> <li>2. El actor introduce los datos por pantalla y pulsa el botón "Enviar"</li> <li>3. Se va al caso de uso "Pantalla Principal"</li> </ol>
<b>Flujo Alternativo</b>	Volver a la página de Login

<b>Caso de Uso</b>	Pantalla Principal
<b>Descripción</b>	Permite al Usuario seleccionar la acción deseada.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ Usuario logado en el sistema</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Se visualiza una pantalla de inicio, con las opciones a seleccionar.</li> <li>2. El actor selecciona la acción:               <ol style="list-style-type: none"> <li>a. Proyectos</li> <li>b. Empleados</li> <li>c. Alarmas</li> <li>d. Informes</li> <li>e. Administración</li> </ol> </li> <li>3. Se va al caso de uso seleccionado</li> </ol>
<b>Flujo Alternativo</b>	Volver a la página de inicio

### 5.2.3.2 - Proyectos

<b>Caso de Uso</b>	Menú de Proyectos
<b>Descripción</b>	Permite al Usuario seleccionar la acción deseada sobre los proyectos.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ Usuario logado en el sistema</li> <li>▪ Usuario con permisos, para acceder a la funcionalidad, según su perfil.</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Se visualiza una pantalla de inicio, con las opciones a seleccionar.</li> <li>2. El actor selecciona la acción: <ol style="list-style-type: none"> <li>a. Nuevo Proyecto</li> <li>b. Modificar Proyecto</li> <li>c. Eliminar Proyecto</li> <li>d. Buscar Proyectos</li> <li>e. Cerrar Proyecto</li> </ol> </li> <li>3. Se va al caso de uso seleccionado</li> </ol>
<b>Flujo Alternativo</b>	Volver a la página de inicio

<b>Caso de Uso</b>	Crear Proyecto
<b>Descripción</b>	Permite al Usuario dar de Alta un nuevo proyecto en el sistema, indicando descripción, herramientas utilizadas en el mismo, y pudiendo autoasignar los empleados a participar en el resto de fases
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a gestionar proyectos, atendiendo a su perfil</li> <li>▪ Proyecto No creado en el sistema</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>▪ Proyecto creado en el sistema</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de "Nuevo Proyecto", en el Menú "Cartera".</li> <li>2. El actor introduce descripción de proyecto</li> <li>3. Pulsa Botón de "Planificación".</li> <li>4. El Actor Introduce la planificación del proyecto</li> <li>5. Pulsa Botón de "Guardar".</li> <li>6. El Actor Introduce las herramientas utilizadas en el proyecto, y la dificultades estimada. <ol style="list-style-type: none"> <li>a. Pulsa "Añadir Otro", para incluir más herramientas. Vuelve al punto 6.</li> <li>b. Pulsa Botón de "Guardar" para continuar.</li> </ol> </li> <li>7. Se Visualiza una pantalla indicando al actor si desea autoplanificar recursos o no, con dos botones, con las opciones "SI" o "NO" <ol style="list-style-type: none"> <li>a. Si el Actor pulsa "SI", el sistema procesa la petición, y</li> </ol> </li> </ol>

	<p>devuelve una pantalla con los empleados asignados a cada fase, y continua al paso 8.</p> <p>b. Si el Actor pulsa "NO", el proceso continua al paso 8.</p> <p>8. Se muestra una pantalla indicando que el proyecto ha sido creado y almacenado correctamente en el sistema.</p>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Modificar Proyecto
<b>Descripción</b>	Permite al Usuario modificar un nuevo proyecto dado de alta anteriormente en el sistema.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a gestionar proyectos, atendiendo a su perfil</li> <li>▪ Proyecto creado en el sistema con anterioridad</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>▪ Proyecto modificado en el sistema</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de "Modificar Proyecto", en el Menú "Cartera".</li> <li>2. Se visualiza una pantalla para la búsqueda de proyecto, bien mediante "Código", o bien mediante "Nombre de proyecto"</li> <li>3. El actor introduce uno de los dos valores solicitados</li> <li>4. Pulsa Botón de "Buscar". <ol style="list-style-type: none"> <li>a. Si se encuentra el proyecto, aparece la descripción del proyecto para ser modificada y continua el caso de uso</li> <li>b. Si no se encuentra el proyecto, aparece un mensaje de error, y se vuelve al paso 2.</li> </ol> </li> <li>5. El Actor modifica ( o no ) el Proyecto y pulsa: <ol style="list-style-type: none"> <li>a. "Guardar y Salir": Se guardan las modificaciones de la descripción, y se finaliza el caso de uso</li> <li>b. "Guardar y Planificación": Se guardan las modificaciones de la descripción y se continua el caso de uso.</li> </ol> </li> <li>6. Se visualiza la planificación del proyecto para ser modificada</li> <li>7. El Actor modifica ( o no ) la planificación y pulsa: <ol style="list-style-type: none"> <li>a. "Guardar y Salir": Se guardan las modificaciones en la planificación, y se finaliza el caso de uso</li> <li>b. "Guardar y Conocimientos": Se guardan las modificaciones en la planificación y se continua el caso de uso.</li> </ol> </li> <li>8. Se visualizan los conocimientos necesarios para el proyecto para ser modificados.</li> <li>9. El Actor tiene tres opciones: <ol style="list-style-type: none"> <li>a. Selecciona los conocimientos a eliminar, y pulsa el botón "Eliminar Conocimiento", con lo que se elimina este de BBDD, y se vuelve al paso 9:</li> <li>b. Pula "Introducir Nuevo Conocimiento":</li> </ol> </li> </ol>

	<p>i. El Actor Introduce las herramientas utilizadas en el proyecto, y la dificultad estimada.</p> <ol style="list-style-type: none"> <li>1. Pulsa "Añadir Otro", para incluir más herramientas volviendo al 10.b</li> <li>2. Pulsa Botón de "Guardar" para volver al paso 9.</li> </ol> <p>c. "Guardar": Se guardan las modificaciones en los conocimientos y se continua el caso de uso.</p> <p>10. Se muestra una pantalla indicando que el proyecto ha sido modificado y almacenado correctamente en el sistema.</p>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Subir Documentación Proyecto
<b>Descripción</b>	Permite al Usuario subir al repositorio documentación relativa al proyecto
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a gestionar proyectos, atendiendo a su perfil</li> <li>▪ Proyecto creado en el sistema con anterioridad</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>▪ Documentación subida al repositorio del proyecto</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de "Modificar Proyecto", en el Menú "Cartera".</li> <li>2. Se visualiza una pantalla para la búsqueda de proyecto, bien mediante "Código", o bien mediante "Nombre de proyecto"</li> <li>3. El actor introduce uno de los dos valores solicitados</li> <li>4. Pulsa Botón de "Buscar". <ol style="list-style-type: none"> <li>a. Si se encuentra el proyecto, aparece la descripción del proyecto y continua el caso de uso</li> <li>b. Si no se encuentra el proyecto, aparece un mensaje de error, y se vuelve al paso 2.</li> </ol> </li> <li>5. El Actor pulsa sobre SubirDocumentación</li> <li>6. Se visualiza la pantalla de selección de documentación</li> <li>7. El Actor selecciona el documento y pulsa sobre el botón "Subir"</li> <li>8. Se muestra una pantalla indicando que la documentación ha sido correctamente subida al servidor.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio



<b>Caso de Uso</b>	Eliminar Proyecto
<b>Descripción</b>	Permite al Usuario eliminar un proyecto dado de alta anteriormente en el sistema.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a gestionar proyectos, atendiendo a su perfil</li> <li>▪ Proyecto creado en el sistema con anterioridad</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>▪ Proyecto dado de baja en el sistema</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de “Eliminar Proyecto”, en el Menú “Cartera”.</li> <li>2. Se visualiza una pantalla para la búsqueda de proyecto, bien mediante “Código”, o bien mediante “Nombre de proyecto”</li> <li>3. El actor introduce uno de los dos valores solicitados</li> <li>4. Pulsa Botón de “Buscar”.               <ol style="list-style-type: none"> <li>a. Si se encuentra el proyecto, aparece la descripción del proyecto para ser modificada y continua el caso de uso</li> <li>b. Si no se encuentra el proyecto, aparece un mensaje de error, y se vuelve al paso 2.</li> </ol> </li> <li>5. Se visualiza la descripción del proyecto a eliminar, y se dan dos opciones:               <ol style="list-style-type: none"> <li>a. “Eliminar”: Se elimina el proyecto de la BBDD y se continua el caso de uso</li> <li>b. “Volver”. Se finaliza el caso de uso</li> </ol> </li> <li>6. Se muestra una pantalla indicando que el proyecto ha sido modificado y almacenado correctamente en el sistema.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Buscar Proyecto
<b>Descripción</b>	Permite al Usuario buscar un proyecto dado de alta anteriormente en el sistema.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a gestionar proyectos, atendiendo a su perfil</li> <li>▪ Proyecto creado en el sistema con anterioridad</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de “Buscar Proyecto”, en el Menú “Cartera”.</li> <li>2. Se visualiza una pantalla para la búsqueda de proyecto, bien mediante “Código”, o bien mediante “Nombre de proyecto”</li> <li>3. El actor introduce uno de los dos valores solicitados</li> <li>4. Pulsa Botón de “Buscar”.               <ol style="list-style-type: none"> <li>a. Si se encuentra el proyecto, se visualiza la descripción del proyecto y continua el caso de uso</li> <li>b. Si no se encuentra el proyecto, aparece un mensaje de error, y</li> </ol> </li> </ol>

	<p style="text-align: center;">se vuelve al paso 2.</p> <p>5. El actor pulsa "Volver", y finaliza el caso de uso</p>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Cerrar Proyecto
<b>Descripción</b>	Permite al dar por cerrar un proyecto dado de alta anteriormente en el sistema, y evaluar el trabajo desempeñado en el mismo.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a gestionar proyectos, atendiendo a su perfil</li> <li>▪ Proyecto creado en el sistema con anterioridad</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>▪ Proyecto Cerrado</li> <li>▪ Empleados Evaluados</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de "Cerrar Proyecto", en el Menú "Cartera".</li> <li>2. Se visualiza una pantalla para la búsqueda de proyecto, bien mediante "Código", o bien mediante "Nombre de proyecto"</li> <li>3. El actor introduce uno de los dos valores solicitados</li> <li>4. Pulsa Botón de "Buscar".               <ol style="list-style-type: none"> <li>a. Si se encuentra el proyecto, y se continua el caso de uso</li> <li>b. Si no se encuentra el proyecto, aparece un mensaje de error, y se vuelve al paso 2.</li> </ol> </li> <li>5. Se visualiza una pantalla, con los datos básicos del proyecto, y desplegados, para indicar la satisfacción del cliente y si ha sido finalizado en fechas. También aparecen el nombre de los empleados, y un desplegable para evaluarlos individualmente.</li> <li>6. El actor selecciona los valores adecuados en los desplegados, y pulsa el botón "Finalizar"</li> <li>7. Se almacenan los valores en BBDD, y aparece una pantalla indicando que el proyecto ha sido cerrado con éxito.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

### 5.2.3.3 - Empleados

<b>Caso de Uso</b>	Empleados
<b>Descripción</b>	Permite al Usuario seleccionar la acción deseada sobre sus datos en el sistema.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ Usuario logado en el sistema</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Se visualiza una pantalla de inicio, con las opciones a seleccionar.</li> <li>2. El actor selecciona la acción: <ol style="list-style-type: none"> <li>a. Ver Datos Propios</li> <li>b. Proyectos Asignados</li> <li>c. Reporte de Horas</li> <li>d. Solicitudes</li> </ol> </li> <li>3. Se va al caso de uso seleccionado</li> </ol>
<b>Flujo Alternativo</b>	Volver a la página de inicio

<b>Caso de Uso</b>	Datos Empleado en sistema
<b>Descripción</b>	Permite al usuario que se encuentra validado en el sistema, ver los datos que hay almacenados en el sistema sobre él, y cambiar las password de acceso, si lo necesita.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de "Ver datos propios", en el Menú "Empleados".</li> <li>2. Se visualiza un pantalla con los datos del usuario logado en el sistema.</li> <li>3. El actor puede: <ol style="list-style-type: none"> <li>a. Pulsa Botón de "Cambiar Password" y el sistema va al caso de uso "Cambiar Password".</li> <li>b. Pulsa el botón "OK"</li> </ol> </li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Cambiar Password
<b>Descripción</b>	Permite al usuario que se encuentra validado en el sistema, cambiar la password con la que entra en el sistema
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> </ul>
<b>Postcondiciones</b>	Password modificada
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de "Cambiar Password", en el Menú "Ver Datos Empleado".</li> <li>2. Se muestra una pantalla para que el usuario introduzca su password antigua, y la password nueva dos veces.</li> <li>3. Pulsa el Botón "Cambiar"</li> <li>4. Se muestra un mensaje indicando que la password ha sido correctamente cambiada.</li> <li>5. Pulsa el botón "OK"</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Proyectos Asignados
<b>Descripción</b>	Permite al Usuario que se encuentra registrado en el sistema, ver qué proyectos tiene actualmente asignados.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de "Proyectos Asignados", en el Menú "Empleados".</li> <li>2. Se visualiza una pantalla para el listado de proyectos que tiene el usuario asignados, la fase en la que lo tiene asignado, fecha inicio, fecha fin asignación</li> <li>3. Pulsa Botón de "Volver".</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Reporte de Horas
<b>Descripción</b>	Permite al Usuario realizar el reporte diario de horas
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>El Usuario debe haberse logado correctamente en el sistema</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>Pulsar el botón de “Reporte de Horas”, en el Menú “Empleados”.</li> <li>Se visualiza una pantalla para la inserción de los datos del reporte</li> <li>El actor puede: <ol style="list-style-type: none"> <li>Introducir los datos solicitados, y pulsar el botón Guardar, y se finaliza el caso de uso.</li> <li>Introducir los datos de reporte y pulsar “Imputar Otro”, para imputar otro día. Se vuelve al paso 2.</li> <li>Pulsar el botón “Ver Imputación” y se va al caso de uso “Consulta de Reportes”</li> </ol> </li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Consulta de Reportes
<b>Descripción</b>	Permite al Usuario realizar el reporte diario de horas, y ver las horas imputadas en días anteriores.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>El Usuario debe haberse logado correctamente en el sistema</li> <li>El Usuario debe de venir del caso de Uso “Reporte de Horas”</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>Pulsar el botón “Ver Imputación” en el menú de “Reporte de Horas”</li> <li>Aparece una nueva pantalla con las imputaciones de los últimos 7 días.</li> <li>Pulsar el botón “Volver”, y se finaliza el caso de uso</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Solicitudes
<b>Descripción</b>	Permite al Usuario gestionar toda la funcionalidad de solicitudes.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>El Usuario debe haberse logado correctamente en el sistema</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>Pulsar el botón de "Solicitudes", en el Menú "Empleados".</li> <li>Se visualiza una pantalla con las opciones: <ol style="list-style-type: none"> <li>"Crear Solicitud": Se va al caso de Uso "Crear Solicitud"</li> <li>"Ver Solicitud": Se va al caso de uso "Consultar Solicitudes".</li> </ol> </li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Crear Solicitud
<b>Descripción</b>	Permite al Usuario crear una nueva solicitud.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>El Usuario debe haberse logado correctamente en el sistema</li> </ul>
<b>Postcondiciones</b>	Solicitud creada
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>Pulsar el botón de "Crear Solicitud", en el Menú "Solicitudes".</li> <li>Se visualiza una pantalla con casillas para ser rellenadas por el usuario.</li> <li>El actor introduce los datos por pantalla, y pulsa "Enviar Solicitud"</li> <li>Se visualiza una pantalla indicando que la solicitud ha sido enviada correctamente.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Ver Solicitudes
<b>Descripción</b>	Permite al Usuario ver las solicitudes que ha lanzado/recibido.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ Para ver las solicitudes Recibidas, el usuario debe de tener los permisos necesarios según su perfil.</li> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de “Ver Solicitudes”, en el Menú “Solicitudes”.</li> <li>2. Se visualiza una pantalla con la lista de las solicitudes lanzadas por el usuario logado.</li> <li>3. El actor tiene dos opciones: <ol style="list-style-type: none"> <li>a. Pulsar botón “OK”, y se finaliza el caso de uso.</li> <li>b. Pulsar botón “Ver Solicitudes Recibidas”, y vamos al caso de uso “Aprobar Solicitudes”</li> </ol> </li> <li>4. Se visualiza una pantalla indicando que la solicitud ha sido enviada correctamente.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Aprobar Solicitudes
<b>Descripción</b>	Permite al Usuario aprobar las solicitudes que le han enviado.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El usuario debe de tener los permisos necesarios según su perfil para aprobar solicitudes.</li> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>▪ Solicitud Aprobada/rechazada</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de “Ver Solicitudes Recibidas”, en el Menú “Ver Solicitudes”.</li> <li>2. Se visualiza una pantalla con la lista de las solicitudes recibidas por el usuario logado, junto con un check box en cada una de ellas.</li> <li>3. El actor selecciona las solicitudes a aprobar y pulsa el botón “Aprobar Seleccionadas”</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

### 5.2.3.4 - Alarmas

<b>Caso de Uso</b>	Alarmas
<b>Descripción</b>	Permite al Usuario seleccionar la acción deseada sobre alarmas
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ Usuario logado en el sistema</li> <li>▪ El Usuario ha de estar autorizado a ver planificación de informes, atendiendo a su perfil</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Se visualiza una pantalla de inicio, con las opciones a seleccionar.</li> <li>2. El actor selecciona la acción: <ol style="list-style-type: none"> <li>a. Crear Alarma</li> <li>b. Modificar Alarma</li> <li>c. Borrar Alarma</li> <li>d. Mostrar Alarmas</li> </ol> </li> <li>3. Se va al caso de uso seleccionado</li> </ol>
<b>Flujo Alternativo</b>	Volver a la página de inicio

<b>Caso de Uso</b>	Crear Alarma
<b>Descripción</b>	Permite al Usuario dar de Alta una nueva alarma en el sistema, indicando su nombre, tipo, condiciones, etc.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a gestionar alarmas, atendiendo a su perfil</li> <li>▪ Alarma no creada en el sistema</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>▪ Alarma creada en el sistema</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de “Crear Alarma”, en el Menú “Alarmas”.</li> <li>2. El actor introduce el nombre de la alarma, tipo, condiciones, etc.</li> <li>3. Pulsa Botón de “Guardar”.</li> <li>4. Se muestra una pantalla indicando que la alarma ha sido creada y almacenada correctamente en el sistema.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio



<b>Caso de Uso</b>	Modificar Alarma
<b>Descripción</b>	Permite al Usuario modificar una alarma existente en el sistema.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a gestionar alarmas, atendiendo a su perfil</li> <li>▪ Alarma creada en el sistema</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>▪ Alarma modificada en el sistema</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de “Modificar Alarma”, en el Menú “Alarmas”.</li> <li>2. Se visualiza una pantalla para la búsqueda de la alarma, con un desplegable con el nombre de todas las alarmas creadas.</li> <li>3. El actor pulsa Botón de “Buscar”.</li> <li>4. Se visualiza una pantalla con los datos actuales de la alarma</li> <li>5. El actor introduce las modificaciones necesarias</li> <li>6. Pulsa Botón de “Guardar”.</li> <li>7. Se muestra una pantalla indicando que la alarma ha sido modificada y almacenada correctamente en el sistema.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Eliminar Alarma
<b>Descripción</b>	Permite al Usuario eliminar una alarma existente en el sistema.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a gestionar alarmas, atendiendo a su perfil</li> <li>▪ Alarma creada en el sistema</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>▪ Alarma eliminada del sistema</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de “Eliminar Alarma”, en el Menú “Alarmas”.</li> <li>2. Se visualiza una pantalla para la búsqueda de la alarma, con un desplegable con el nombre de todas las alarmas creadas.</li> <li>3. El actor pulsa Botón de “Buscar”.</li> <li>4. Se visualiza una pantalla con los datos de la alarma a eliminar</li> <li>5. El actor pulsa el Botón de “Eliminar”.</li> <li>6. Se muestra una pantalla indicando que la alarma ha sido eliminada correctamente del sistema.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Mostrar Alarmas
<b>Descripción</b>	Permite al Usuario ver un listado de las alarmas existentes en el sistema.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"><li>▪ El Usuario debe haberse logado correctamente en el sistema</li><li>▪ El Usuario ha de estar autorizado a gestionar alarmas, atendiendo a su perfil</li></ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"><li>1. Pulsar el botón de “Mostrar Alarmas”, en el Menú “Alarmas”.</li><li>2. Se visualizan todas las alarmas existentes.</li></ol>
<b>Flujo Alternativo</b>	Volver al Inicio

### 5.2.3.5 - Informes

<b>Caso de Uso</b>	Informes
<b>Descripción</b>	Permite al Usuario seleccionar la acción deseada sobre los informes
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ Usuario logado en el sistema</li> <li>▪ El Usuario ha de estar autorizado a ver planificación de informes, atendiendo a su perfil</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Se visualiza una pantalla de inicio, con las opciones a seleccionar.</li> <li>2. El actor selecciona la acción: <ol style="list-style-type: none"> <li>a. Crear Nuevo Informe</li> <li>b. Informes Favoritos</li> <li>c. Planificar Informes</li> <li>d. Ver Planificación</li> <li>e. Repositorio de Informes</li> </ol> </li> <li>3. Se va al caso de uso seleccionado</li> </ol>
<b>Flujo Alternativo</b>	Volver a la página de inicio

<b>Caso de Uso</b>	Crear Nuevo Informe
<b>Descripción</b>	Permite al Usuario seleccionar los objetos, con los que desea generar un nuevo informe. El sistema transformará dichos objetos a una sql entendible por la BBDD, para ejecutar dicho informe y devolver los datos al usuario.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a crear informes, atendiendo a su perfil</li> </ul>
<b>Postcondiciones</b>	Informe Generado
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de "Crear Nuevo Informe", en el Menú "Informes".</li> <li>2. Se visualiza una pantalla con todos los objetos disponibles.</li> <li>3. El actor selecciona los objetos que quiere que aparezcan en el informe, y establece los filtros que desee aplicar al mismo.</li> <li>4. Pulsa Botón de "Ejecutar".</li> <li>5. Se ejecuta la sql generada con los objetos seleccionados, y muestra una pantalla con el informe resultado.</li> <li>6. Pulsar Botón de: <ol style="list-style-type: none"> <li>a. "Guardar como favorito" :El sistema se va al caso de uso "Crear Favorito".</li> <li>b. "Exportar a Excel" : El informe se exporta a un excel que es depositado en el repositorio del usuario.</li> <li>c. "Volver": Se vuelve al menú principal de Informes.</li> </ol> </li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Crear Favorito
<b>Descripción</b>	Permite al Usuario guardar un informe generado con anterioridad como favorito, para poder acceder a él posteriormente.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>El Usuario debe haberse logado correctamente en el sistema</li> <li>El Usuario ha de estar autorizado a crear informes, atendiendo a su perfil</li> <li>Se debe de haber pasado por el caso de uso "Crear Nuevo Informe"</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>Informe Guardado como Favorito</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>Pulsar el botón de "Guardar como favorito" : El informe se almacena en BBDD como Favorito, para que el usuario pueda volver a consultarlo en un futuro.</li> <li>"Volver": Se vuelve al menú principal de Informes.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Informes favoritos
<b>Descripción</b>	Permite al obtener los informes que previamente ha indicado como Favoritos, para volver a visualizarlos.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>El Usuario debe haberse logado correctamente en el sistema</li> <li>El Usuario ha de estar autorizado a gestionar informes, atendiendo a su perfil</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>Pulsar el botón de "Informes Favoritos", en el Menú "Informes".</li> <li>Se visualiza una pantalla con el listado de informes disponibles ( Nombre de Informe, Fecha de creación), y un check para seleccionar el deseado.</li> <li>El actor pulsa el check del informe deseado y pulsa "Ejecutar".</li> <li>El sistema ejecuta de la sql, y se visualiza una pantalla con el resultado del informe.</li> <li>Pulsa Botón de "Volver" y se devuelve al usuario al menú de "Informes".</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Eliminar Informes favoritos
<b>Descripción</b>	Permite al eliminar un informe que previamente ha indicado como Favoritos.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>El Usuario debe haberse logado correctamente en el sistema</li> <li>El Usuario ha de estar autorizado a gestionar informes, atendiendo a su perfil</li> </ul>
<b>Postcondiciones</b>	Informe eliminado del sistema
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>Pulsar el botón de “Informes Favoritos”, en el Menú “Informes”.</li> <li>Se visualiza una pantalla con el listado de informes disponibles ( Nombre de Informe, Fecha de creación), y un check para seleccionar el deseado.</li> <li>El actor pulsa el check del informe deseado y pulsa “Eliminar Favorito”.</li> <li>Aparece un mensaje por pantalla indicando que el informe ha sido correctamente dado de baja del sistema</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Planificar Informes
<b>Descripción</b>	Permite al usuario, seleccionar la periodicidad con la que quiere que se actualicen los datos de un informe, Fecha, Hora, etc, y la ruta donde desea guardar el resultado del mismo, para poder acceder a los datos.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>El Usuario debe haberse logado correctamente en el sistema</li> <li>El Usuario ha de estar autorizado a planificar informes, atendiendo a su perfil</li> <li>Alarma creada en el sistema</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>Informe correctamente planificado.</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>Pulsar el botón de “Planificar Informes”, en el Menú “Informes”.</li> <li>Se visualiza una pantalla para la introducción de los datos del informe. Se introduce Nombre, Tipo planificación hora y día y se pulsa el botón "Guardar"</li> <li>Se visualiza una pantalla indicando que el informe ha sido planificado correctamente.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Ver Planificación
<b>Descripción</b>	Permite al Usuario ver un listado de los informes planificados en el sistema, y un link a la ruta de los mismos.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a ver planificación de informes, atendiendo a su perfil</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de “Ver Planificación”, en el Menú “Informes”.</li> <li>2. Se visualizan todos los informes planificados.</li> <li>3. El usuario puede: <ol style="list-style-type: none"> <li>a. Seleccionar el link del informe que desee, lo que le llevará a la ruta del mismo.</li> </ol> </li> <li>4. El Usuario pulsa “Volver” y el sistema le devuelve a la página de inicio de “Informes”.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Repositorio de Informe
<b>Descripción</b>	Permite al Usuario ver un listado de los informes que han sido generados, bien por petición manual, o bien mediante la ejecución del informe planificado.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado para acceder a su repositorio</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de “Repositorio de Informes”, en el Menú “Informes”.</li> <li>2. Se visualizan todos los informes generados.</li> <li>3. El usuario puede: <ol style="list-style-type: none"> <li>a. Descargar el informe en formato excel, pulsando el icono de Excel.</li> <li>b. Borrar el informe, pulsando el icono de papelera</li> </ol> </li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

### 5.2.3.6 - Administración

<b>Caso de Uso</b>	Administración
<b>Descripción</b>	Permite al Usuario seleccionar la acción deseada sobre temas administrativos
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ Usuario logado en el sistema</li> <li>▪ El Usuario ha de estar autorizado a ver planificación de informes, atendiendo a su perfil</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Se visualiza una pantalla de inicio, con las opciones a seleccionar.</li> <li>2. El actor selecciona la acción: <ol style="list-style-type: none"> <li>a. Alta Empleado</li> <li>b. Modificar Datos Empleados</li> <li>c. Dar de Baja Empleado</li> <li>d. Cambiar Derechos</li> </ol> </li> <li>3. Se va al caso de uso seleccionado</li> </ol>
<b>Flujo Alternativo</b>	Volver a la página de inicio

<b>Caso de Uso</b>	Alta Empleado
<b>Descripción</b>	Permite al Usuario-Administrador dar de alta un nuevo empleado en el sistema, indicando descripción (Nombre, Apellido, Login, etc, herramientas con las que el empleado está familiarizado, y su nivel de conocimiento en las mismas.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a Administrar Empleados, atendiendo a su perfil</li> <li>▪ Empleado NO dado de alta en el sistema</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>▪ Empleado dado de alta en el sistema</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de “Alta Empleado”, en el Menú “Administración”.</li> <li>2. Aparece una nueva pantalla, donde el actor introduce los datos relativos al empleado.</li> <li>3. Pulsa Botón de “Continuar”.</li> <li>4. El Actor Introduce las herramientas en las que el empleado posee un conocimiento, y el nivel del mismo. <ol style="list-style-type: none"> <li>a. Pulsa “Añadir Otro”, para incluir más herramientas</li> <li>b. Pulsa Botón de “Guardar” para continuar</li> </ol> </li> <li>5. Se muestra una pantalla indicando que el empleado ha sido dado de alta en el sistema correctamente.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio
<b>Caso de Uso</b>	Modificar Datos Empleado
<b>Descripción</b>	Permite al Usuario-Administrador modificar los datos de un empleado dado de alta

	anteriormente en el sistema.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a administrar los datos de los empleado, atendiendo a su perfil</li> <li>▪ Empleado ya dado de alta en el sistema con anterioridad</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>▪ Empleado modificado en el sistema</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de "Modificar Datos Empleado", en el Menú "Administración".</li> <li>2. Se visualiza una pantalla para la búsqueda de empleados, bien mediante "Login", o bien mediante "Nombre" y "Apellido" de Empleado</li> <li>3. El actor introduce uno de los dos valores solicitados</li> <li>4. Pulsa Botón de "Buscar". <ol style="list-style-type: none"> <li>a. Si se encuentra el empleado, aparecen sus datos y continua el caso de uso</li> <li>b. Si no se encuentra el empleado, aparece un mensaje de error, y se vuelve al paso 2.</li> </ol> </li> <li>5. El Actor modifica (o no) los datos del empleado.</li> <li>6. Pulsa botón de "Continuar".</li> <li>7. Aparece una pantalla con los conocimientos que posee el empleado para ser modificados.</li> <li>8. El Actor tiene tres opciones: <ol style="list-style-type: none"> <li>a. Selecciona los conocimientos a eliminar, y pulsa el botón "Eliminar Conocimiento", con lo que se elimina este de BBDD, y se vuelve al paso 9:</li> <li>b. Pula "Introducir Nuevo Conocimiento": <ol style="list-style-type: none"> <li>i. El Actor Introduce las herramientas que conoce el empleado, y el nivel del mismo. <ol style="list-style-type: none"> <li>1. Pulsa "Añadir Otro", para incluir más herramientas volviendo al 10.b</li> <li>2. Pulsa Botón de "Guardar" para volver al paso 9.</li> </ol> </li> <li>c. "Guardar": Se guardan las modificaciones en los conocimientos y se continua el caso de uso.</li> </ol> </li> </ol> </li> <li>9. Se muestra una pantalla indicando que el empleado ha sido modificado correctamente en el sistema.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio



<b>Caso de Uso</b>	Dar de Baja Empleado
<b>Descripción</b>	Permite al Usuario dar de baja a un empleado dado de alta anteriormente en el sistema.
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a gestionar empleados, atendiendo a su perfil</li> <li>▪ Empleado creado en el sistema con anterioridad</li> </ul>
<b>Postcondiciones</b>	<ul style="list-style-type: none"> <li>▪ Empleado dado de baja en el sistema</li> </ul>
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de “Dar de Baja Empleado”, en el Menú “Administración”.</li> <li>2. Se visualiza una pantalla para la búsqueda de empleados, bien mediante “Login”, o bien mediante “Nombre” y “Apellido” de Empleado</li> <li>3. El actor introduce uno de los dos valores solicitados</li> <li>4. Pulsa Botón de “Buscar”.               <ol style="list-style-type: none"> <li>a. Si se encuentra el empleado, aparecen sus datos y continua el caso de uso</li> <li>b. Si no se encuentra el empleado, aparece un mensaje de error, y se vuelve al paso 2.</li> </ol> </li> <li>5. Se visualiza una pantalla con los datos principales del usuario a dar de baja.</li> <li>6. Pulsa botón de “Eliminar”</li> <li>7. Se muestra una pantalla indicando que el empleado ha sido dado de baja correctamente en el sistema.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Cambiar Derechos
<b>Descripción</b>	Permite al Usuario-Administrador cambiar los derechos que tiene con respecto a las distintas funcionalidades del sistema
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> <li>▪ El Usuario ha de estar autorizado a cambiar derechos del sistema, atendiendo a su perfil</li> <li>▪ Proyecto creado en el sistema con anterioridad</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Pulsar el botón de “Cambiar Derechos”, en el Menú “Administración”.</li> <li>2. Se visualiza una pantalla con los distintos perfiles disponibles</li> <li>3. El actor introduce “pincha” en el perfil a modificar.</li> <li>4. Se visualiza una pantalla con los distintos permisos que se puede otorgar a cada tipo de empleado, y marcados los “check” que tiene activos.</li> <li>5. El actor, selecciona o deselecciona, los check del perfil y pulsa el botón “Validar”</li> <li>6. Aparece una pantalla indicando que los permisos han sido modificados correctamente.</li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

<b>Caso de Uso</b>	Mensajes en Sistema
<b>Descripción</b>	Permite al Usuario ver los mensajes que tiene en el servidor interno de la aplicación
<b>Actores</b>	Usuario
<b>Precondiciones</b>	<ul style="list-style-type: none"> <li>▪ El Usuario debe haberse logado correctamente en el sistema</li> </ul>
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Desde cualquier página de la aplicación, el usuario clics sobre el icono de "sobre".</li> <li>2. Se visualiza una pantalla con los mensajes del sistema, indicando si son nuevos, ya leídos.</li> <li>8. El actor puede               <ol style="list-style-type: none"> <li>c. Marcar un mensaje como NoLeído, seleccionando el botón de "Marcar como No Leído"</li> <li>d. Marcar un mensaje como Leído, seleccionando el botón de "Marcar como Leído"</li> <li>e. Eliminar un mensaje, seleccionando el botón "Eliminar"</li> </ol> </li> </ol>
<b>Flujo Alternativo</b>	Volver al Inicio

## 5.3 - Modelo de Datos:

### 5.3.1 - Introducción

Para la correcta utilización de la aplicación, y el almacenamiento de la información, se ha optado por el almacenamiento dentro de un gestor de base de datos Mysql, dentro del cual ha sido creada una base de datos con nombre "Gestion", y dentro de la cual se crean las distintas tablas dentro de las que se guardará la información del sistema.

Dentro del siguiente apartado, se muestran las distintas tablas que han sido creadas dentro de esta base de datos, así como se detalla el significado de cada uno de los campos que estas contienen.

### 5.3.2 - Modelado de la Base de Datos

#### ➤ Alarmas

```
| Alarmas | CREATE TABLE `Alarmas` (  
  `Nombre_Alarma` varchar(30) NOT NULL,  
  `Tipo_Aviso` varchar(20) NOT NULL,  
  `Campo_Condicion` varchar(22) NOT NULL,  
  `Condicion` tinyint(4) NOT NULL,  
  `Perfil_Afectado` varchar(19) NOT NULL,  
  `Fecha_Ult_Modificacion` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
  `Borrado` tinyint(4) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 7 - DLL tabla Alarmas

#### Descripción de Campos:

- **Nombre\_Alarma** --> Nombre único de Alarma dentro de BBDD
- **Tipo\_Aviso** --> --> Indica el aviso que generará la alarma una vez cumpla con las condiciones de ejecución. Los posibles valores son Correo, Mensaje en Sistema, Aviso Responsable Campo\_Condición --> Indica el campo condición que afecta a la alarma. Este campo condición será el inicio/fin de cada una de las fases del proyecto. Ej., Ini\_Aceptación, Fin\_Oferta, etc.
- **Condición** --> Indica el número de días que han de quedar/pasar para el cumplimiento de la condición. Valores de -7 a 7. Por ejemplo, un valor -2 para Campo Condición Ini\_Oferta, significa que la alarma saltará dos días antes de que se cumpla la fecha de inicio de la fase de Oferta.
- **Perfil\_Afectado** --> Indica los empleados afectados por la alarma según perfil .
- **Fecha\_Ult\_Modificacion** --> Fecha de Última modificación del registro
- **Borrado** --> Indica si la Alarma está activa o no.

➤ Calendar

```
| Calendar | CREATE TABLE `Calendar` (  
  `DateField` date NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 |
```

Figura 8 - DLL tabla Calendar

Descripción de Campos:

- **Fecha** --> Campo con un valor Fecha

➤ Cat\_Condiciones Alarmas

```
| Cat_Condiciones_Alarmas | CREATE TABLE `Cat_Condiciones_Alarmas` (  
  `Condicion` varchar(22) NOT NULL,  
  `Borrado` tinyint(4) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 9 - DLL tabla Cat\_Condiciones\_Alarmas

Descripción de Campos:

- **Condición** --> Valores de las posibles condiciones que puede tener una alarma
- **Borrado** --> Indica si la condición está activa o no.

➤ Cat\_Equipos

```
| Cat_Equipos | CREATE TABLE `Cat_Equipos` (  
  `Equipo` varchar(25) NOT NULL,  
  `Fecha_Alta` date NOT NULL,  
  `Borrado` tinyint(4) NOT NULL,  
  `Responsable` varchar(10) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 10 - DLL tabla Cat\_Equipos

Descripción de Campos:

- **Equipo** --> Nombre del Equipo dado de alta en el sistema
- **Fecha\_Alta** --> Fecha en la que se dio de alta el equipo en el sistema.
- **Borrado** --> Indica si el Equipo está dado de alta o no.
- **Responsable** --> Nombre del responsable del equipo

➤ **Cat Fases**

```
| Cat_Fases | CREATE TABLE `Cat_Fases` (  
  `Fase` varchar(25) NOT NULL,  
  `Borrado` tinyint(4) NOT NULL,  
  `Perfil` varchar(15) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 11 - DLL tabla Cat\_Fases

Descripción de Campos:

- **Fase** --> Fase posible para los proyectos en el sistema
- **Borrado** --> Indica si la fase está dada de alta o no en el sistema
- **Perfil** --> Perfil capacitado para llevar a cabo dicha fase.

➤ **Cat Herramientas**

```
| Cat_Herramientas | CREATE TABLE `Cat_Herramientas` (  
  `Herramienta` varchar(50) NOT NULL,  
  `Borrado` tinyint(4) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 12 - DLL tabla Cat\_Herramientas

Descripción de Campos:

- **Herramienta** --> Nombre de la herramienta dada de alta en el sistema
- **Borrado** --> Indica si la herramienta está dada de alta en el sistema o no.

➤ **Cat Releases**

```
| Cat_Releases | CREATE TABLE `Cat_Releases` (  
  `Cod_Release` varchar(8) NOT NULL,  
  `Borrado` tinyint(4) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 13 - DLL tabla Cat\_Releases

Descripción de Campos:

- **Cod\_Release** --> *Códigos de las releases disponibles*
- **Borrado** --> *Indica si la release está dada de alta en el sistema o no.*

➤ Cat Tipos Alarmas

```
| Cat_Tipos_Alarmas | CREATE TABLE `Cat_Tipos_Alarmas` (  
  `Tipo_Alarma` varchar(20) NOT NULL,  
  `Borrado` tinyint(4) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 14 - DLL tabla Cat\_Tipos\_Alarmas

Descripción de Campos:

- **Tipo\_Alarma** -->Tipos de Alarmas disponibles en el sistema
- **Borrado** -->Indica si el tipo de alarma está dada de alta en el sistema o no.

➤ Conocimientos Empleados

```
| Conocimientos_Empleados | CREATE TABLE `Conocimientos_Empleados` (  
  `Usuario` varchar(10) NOT NULL,  
  `Conocimiento` varchar(50) NOT NULL,  
  `Nivel_Conocimiento` varchar(7) NOT NULL,  
  `Borrado` tinyint(4) NOT NULL,  
  `Fecha_Carga` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 15 - DLL tabla Cat\_Condiciones\_Alarmas

Descripción de Campos:

- **Usuario** --> Usuario que posee el conocimiento
- **Conocimiento** --> Herramienta de la que el empleado tiene conocimiento
- **Nivel** --> Nivel del conocimiento seleccionado
- **Borrado** --> Indica si el conocimiento está activo o no
- **Fecha\_Carga** -->Fecha en la que se ha modificado por última vez el registro en BBDD.

➤ **Conocimientos Proyecto**

```
| Conocimientos_Proyecto | CREATE TABLE `Conocimientos_Proyecto` (  
  `Codigo_Proyecto` varchar(10) NOT NULL,  
  `Conocimiento` varchar(50) NOT NULL,  
  `Nivel_Conocimiento` varchar(7) NOT NULL,  
  `Borrado` tinyint(4) NOT NULL,  
  `Fecha_Carga` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 16 - DDL tabla Conocimientos\_Proyecto

**Descripción de Campos:**

- **Codigo\_Proyecto** --> Código del proyecto que requiere el conocimiento
- **Conocimiento** --> Herramienta de la que el proyecto requiere un conocimiento
- **Nivel** --> Nivel del conocimiento seleccionado
- **Borrado** --> Indica si el conocimiento está activo o no
- **Fecha\_Carga** --> Fecha en la que se ha modificado por última vez el registro en BBDD.

➤ Derechos

```

| Derechos | CREATE TABLE `Derechos` (
  `Perfil` varchar(30) NOT NULL,
  `Car_Nuevo_Proyecto` tinyint(4) NOT NULL,
  `Car_Modificar_Proyecto` tinyint(4) NOT NULL,
  `Car_Eliminar_Proyecto` tinyint(4) NOT NULL,
  `Car_Buscar_Proyecto` tinyint(4) NOT NULL,
  `Car_Cerrar_Proyecto` tinyint(4) NOT NULL,
  `Emp_Ver_Datos_Propios` tinyint(4) NOT NULL,
  `Emp_Proyectos_Asignados` tinyint(4) NOT NULL,
  `Emp_Reporte_Horas` tinyint(4) NOT NULL,
  `Emp_Solicitudes` tinyint(4) NOT NULL,
  `Emp_Aprobar_Solicitudes` tinyint(4) NOT NULL,
  `Alar_Crear_Alarma` tinyint(4) NOT NULL,
  `Alar_Modificar_Alarma` tinyint(4) NOT NULL,
  `Alar_Borrar_Alarma` tinyint(4) NOT NULL,
  `Alar_Mostrar_Alarma` tinyint(4) NOT NULL,
  `Inf_Crear_Nuevo_Informe` tinyint(4) NOT NULL,
  `Inf_Informes_Favoritos` tinyint(4) NOT NULL,
  `Inf_Planificar_Informes` tinyint(4) NOT NULL,
  `Inf_Ver_Planificacion` tinyint(4) NOT NULL,
  `Adm_Alta_Empleado` tinyint(4) NOT NULL,
  `Adm_Modificar_Datos_Empleado` tinyint(4) NOT NULL,
  `Adm_Dar_Baja_Empleado` tinyint(4) NOT NULL,
  `Adm_Cambiar_Derechos` tinyint(4) NOT NULL,
  `Borrado` tinyint(4) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |

```

Figura 17 - DLL tabla Derechos

Descripción de Campos:

- **Perfil** --> Perfil de Empleado al que corresponde el derecho
- **Car\_Nuevo\_Proyecto** --> Indica si tiene derechos para crear un nuevo proyecto
- **Car\_Modificar\_Proyecto** --> Indica si tiene derechos para modificar un proyecto
- **Car\_Eliminar\_Proyecto** --> Indica si tiene derechos para eliminar un proyecto
- **Car\_Buscar\_Proyecto** --> Indica si tiene derechos para buscar un proyecto de lectura
- **Car\_Cerrar\_Proyecto** --> Indica si tiene derechos para cerrar un proyecto
- **Emp\_Ver\_Datos\_Propios** --> Indica si tiene derechos para ver datos propios en la herramienta
- **Emp\_Proyectos\_Asignados** --> Indica si tiene derechos para ver los proyectos que tiene asignados en el sistema
- **Emp\_Reporte\_Horas** --> Indica si tiene derechos para crear un reporte de horas
- **Emp\_Solicitudes** --> Indica si tiene derechos para crear solicitudes en el sistema



- **Emp\_Aprobar\_Solicitudes-->** Indica si tiene derechos para aprobar solicitudes en el sistema
- **Alar\_Crear\_Alarma-->** Indica si tiene derechos para crear alarmas
- **Alar\_Modificar\_Alarma-->** Indica si tiene derechos para modificar alarmas
- **Alar\_Borrar\_Alarma-->** Indica si tiene derechos para eliminar alarmas
- **Alar\_Mostrar\_Alarma-->** Indica si tiene derechos para ver las alarmas del sistema
- **Inf\_Crear\_Nuevo\_Informe-->** Indica si tiene derechos para crear un nuevo informe
- **Inf\_Planificar\_Informes-->** Indica si tiene derechos para planificar un informe
- **Inf\_Ver\_Planificacion -->** Indica si tiene derechos para ver sus informes planificados.
- **Adm\_Alta\_Empleado-->** Indica si tiene derechos para dar de alta un empleado en el sistema
- **Adm\_Modificar\_Datos\_Empleado-->** Indica si tiene derechos para modificar los datos de un empleado
- **Adm\_Dar\_Baja\_Empleado-->** Indica si tiene derechos para dar de baja un empleado
- **Adm\_Cambiar\_Derechos-->** Indica si tiene derechos para cambiar los derechos de los perfiles
- **Borrado -->** Indica si el derecho está activo o no.

➤ **Empleados**

```
| Empleados | CREATE TABLE `Empleados` (  
  `NIF` varchar(9) NOT NULL,  
  `Usuario` varchar(15) NOT NULL,  
  `Nombre` varchar(30) NOT NULL,  
  `Apellidos` varchar(50) NOT NULL,  
  `Domicilio` varchar(50) NOT NULL,  
  `Num_Telefono` int(11) DEFAULT NULL,  
  `Fecha_Nacimiento` date NOT NULL,  
  `Rol` varchar(30) NOT NULL,  
  `Perfil` varchar(30) NOT NULL,  
  `Proyecto_Asignado` varchar(30) NOT NULL,  
  `Borrado` tinyint(4) NOT NULL,  
  `Fecha_Ult_Modificacion` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',  
  `Fecha_Carga` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 |
```

Figura 18 - DDL tabla Empleados

**Descripción de Campos:**

- **NIF** --> NIF del empleado
- **Nombre** --> Nombre del Empleado
- **Apellidos** --> Apellidos del Empleado
- **Domicilio** --> Domicilio de Residencia del Empleado
- **Num\_Telefono** --> Teléfono de contacto del empleado
- **Fecha\_Nacimiento** --> Fecha de nacimiento del empleado
- **Rol** --> Rol del Empleado
- **Perfil** --> Perfil del Empleado
- **Proyecto Asignado** --> Proyecto al que está asignado el empleado
- **Borrado** --> Indica si el empleado está dado de alta en el sistema o no
- **Fecha\_Ult\_Modificacion** --> Fecha de última modificación del registro
- **Fecha\_Carga** --> fecha de carga del registro

➤ **Funcionalidad Pagina**

```
| Funcionalidad_Pagina | CREATE TABLE `Funcionalidad_Pagina` (  
  `Pagina` varchar(50) NOT NULL,  
  `Funcionalidad` varchar(50) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 19 - DLL tabla Funcionalidad\_Pagina

Descripción de Campos:

- **Pagina** --> Página a la que corresponde una funcionalidad
- **Funcionalidad**--> Funcionalidad relativa a la página

➤ **Gestión Errores**

```
| Gestion_Errores | CREATE TABLE `Gestion_Errores` (  
  `Codigo_Error` tinyint(4) NOT NULL,  
  `Descripcion_Error` varchar(250) NOT NULL,  
  `Pagina_Destino` varchar(50) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 20 - DLL tabla Gestion\_Errores

Descripción de Campos:

- **Codigo\_Error** --> Código de error de sistema
- **Descripcion\_Error**--> Descripción del Error
- **Pagina\_Destino** --> Pagina Destino tras mensaje de error

➤ **Gestion OK**

```
| Gestion_OK | CREATE TABLE `Gestion_OK` (  
  `Codigo_Salida` tinyint(4) NOT NULL,  
  `Descripcion_Accion` varchar(250) NOT NULL,  
  `Pagina_Destino` varchar(50) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 21 - DLL tabla Gestion\_OK

Descripción de Campos:

- **Codigo\_Salida**--> Código de OK generado en el sistema
- **Descripcion\_Accion**--> Descripción del OK
- **Pagina\_Destino** --> Pagina destino tras mensaje OK

### ➤ Informes

```
| Informes | CREATE TABLE `Informes` (  
  `Nombre_Informe` varchar(30) NOT NULL,  
  `Fecha_Creacion` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
  `Descripción_Informe` varchar(250) DEFAULT NULL,  
  `Creador` varchar(30) NOT NULL,  
  `Sentencia` varchar(500) NOT NULL,  
  `Borrado` tinyint(4) NOT NULL,  
  `objid` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 22 - DLL tabla Informes

#### Descripción de Campos:

- **Nombre\_Informe** --> Nombre del Informe en el sistema
- **Creador** --> Creador del Informe
- **Sentencia** --> Sentencia sql generada en el informe
- **Fecha\_Creacion** --> Fecha de creación del informe
- **Borrado** --> Indica si el informe está activo o no

### ➤ Login

```
| Login | CREATE TABLE `Login` (  
  `Usuario` varchar(15) NOT NULL,  
  `Password` varchar(15) NOT NULL,  
  `Borrado` tinyint(4) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |  
+-----+
```

Figura 23 - DLL tabla Login

#### Descripción de Campos:

- **Usuario** --> Login del usuario dado de alta en el sistema
- **Password** --> Password del usuario
- **Borrado** --> Indica si el usuario está activo o no

➤ **Mensajes Sistema**

```
| Mensajes_Sistema | CREATE TABLE `Mensajes_Sistema` (  
  `Id_Mensaje` tinyint(4) NOT NULL,  
  `Usuario` varchar(15) NOT NULL,  
  `Asunto` varchar(200) DEFAULT NULL,  
  `Mensaje` varchar(500) DEFAULT NULL,  
  `Fecha_Mensaje` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
  `Borrado` tinyint(4) NOT NULL,  
  `Fecha_Borrado` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',  
  `Leido` tinyint(4) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |  
+-----+
```

Figura 24 - DLL tabla Mensajes\_Sistema

**Descripción de Campos:**

- **Id\_Mensaje** --> Identificador único de mensajes en el sistema
- **Usuario** --> Login del usuario al que pertenece el mensaje
- **Asunto** --> Asunto del mensaje enviado
- **Mensaje** --> Cuerpo del mensaje
- **Fecha\_Mensaje** --> Fecha en la que el mensaje fue enviado
- **Borrado** --> Indica si el mensaje ha sido borrado o no
- **Fecha\_Borrado** --> Fecha en la que el mensaje fue borrado
- **Leido** --> Indica si el mensaje ha sido leído o no

### ➤ Planificacion Informes

```
| Planificacion Informes | CREATE TABLE `Planificacion_Informes` (  
  `Informe_objid` int(11) NOT NULL,  
  `Tipo_Planificacion` varchar(30) NOT NULL,  
  `Hora_Planificacion` time NOT NULL,  
  `Fecha_Ult_Modificacion` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
  `Borrado` tinyint(4) NOT NULL,  
  `dia_plani` varchar(10) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 |
```

Figura 25 - DLL tabla Planificacion\_Informes

#### Descripción de Campos:

- **Informe\_objid** --> Identificador único de informe planificado
- **Tipo\_Planificacion** --> Tipo de planificación (Diaria, Mensual, Semana....)
- **Hora\_Planificacion** --> Hora en la que se planifica un informe
- **Fecha\_Ult\_Modificacion** --> Fecha de última modificación de la planificación
- **Borrado** --> Indica si la planificación está activa
- **Dia\_plani** --> Indica el día de planificación en caso de planificación Mensuales, Semanales, y una vez.

### ➤ Planificacion Proyecto

```
| Planificacion Proyecto | CREATE TABLE `Planificacion_Proyecto` (  
  `Codigo_Proyecto` varchar(30) NOT NULL,  
  `Inicio_Fase` varchar(30) NOT NULL,  
  `Fin_Fase` varchar(30) NOT NULL,  
  `Jornadas_Fase` int(11) NOT NULL,  
  `Fase` varchar(30) NOT NULL,  
  `Borrado` tinyint(4) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 26 - DLL tabla Planificacion\_Proyecto

#### Descripción de Campos:

- **Codigo\_Proyecto** --> Código de proyecto al que pertenece la planificación
- **Inicio\_Fase** --> Fecha de Comienzo de la fase planificada
- **Fin\_Fase** --> Fecha de Finalización de la fase planificada
- **Jornadas\_Fase** --> Jornadas planificadas para la fase
- **Fase** --> Fase planificada
- **Borrado** --> Indica si la planificación está activa o no

➤ **Proyecto**

```
| Proyecto | CREATE TABLE `Proyecto` (  
  `Codigo_Proyecto` varchar(30) NOT NULL,  
  `Nombre_Proyecto` varchar(30) NOT NULL,  
  `Jefe_Proyecto` varchar(30) NOT NULL,  
  `Equipo_Asignado` varchar(30) NOT NULL,  
  `Total_Jornadas` int(11) NOT NULL,  
  `Dificultad` varchar(8) NOT NULL,  
  `Responsable_Local` varchar(30) NOT NULL,  
  `Estado` varchar(10) NOT NULL,  
  `Satisfaccion_Cliente` int(11) DEFAULT NULL,  
  `Borrado` int(11) NOT NULL,  
  `Fecha_Carga` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  `Fase` varchar(15) NOT NULL,  
  `Release_Objetivo` varchar(10) NOT NULL,  
  `Fin_Fechas` varchar(2) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 27 - DLL tabla Proyecto

Descripción de Campos:

- **Codigo\_Proyecto** --> Código del proyecto
- **Nombre\_Proyecto** --> Nombre del proyecto
- **Jefe\_Proyecto** --> Jefe de proyecto asignado
- **Equipo\_Asignado** --> Equipo al que pertenece el proyecto
- **Total\_Jornadas** --> Total Jornadas de las que consta el proyecto
- **Dificultad** --> Dificultad del proyecto
- **Responsable\_Local** --> Responsable Local del proyecto
- **Estado** --> Estado del proyecto
- **Satisfaccion\_Cliente** --> Satisfacción del cliente pro el proyecto
- **Borrado** --> Indica si el proyecto está activo o no
- **Fecha\_Carga** --> Fecha de última modificación del proyecto
- **Fase** --> Fase en la que se encuentra el proyecto
- **Release\_Objetivo** --> Release objetivo de despliegue del proyecto
- **Fin\_Fechas** --> Indica si el proyecto ha acabado en fechas o no

➤ Rel Proyecto Empleado

```
| Rel_Proyecto_Empleado | CREATE TABLE `Rel_Proyecto_Empleado` (  
  `Codigo_Proyecto` varchar(30) NOT NULL,  
  `Usuario` varchar(15) NOT NULL,  
  `Fase` varchar(30) NOT NULL,  
  `Esfuerzo_Asignado` decimal(10,0) NOT NULL,  
  `Evaluacion_Empleado` tinyint(4) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 28 - DLL tabla Rel\_Proyecto\_Empleado

Descripción de Campos:

- **Codigo\_Proyecto** --> Código de proyecto
- **Usuario**--> Login del empleado
- **Fase** --> Fase que tienen en común
- **Esfuerzo\_Asignado** --> Esfuerzo asignado en la fase
- **Evaluacion\_Empleado** --> Evaluación final del empleado

➤ Reporte Horas

```
| Reporte_Horas | CREATE TABLE `Reporte_Horas` (  
  `Usuario` varchar(15) NOT NULL,  
  `Fecha_Imputacion` date NOT NULL,  
  `Codigo_Proyecto` varchar(30) NOT NULL,  
  `Fase` varchar(30) NOT NULL,  
  `Esfuerzo` decimal(10,0) NOT NULL,  
  `Porcentaje_Avance_Fase` decimal(10,0) NOT NULL,  
  `Fecha_Carga` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 29 - DLL tabla Reporte\_Horas

Descripción de Campos:

- **Usuario** --> Login del empleado que realiza la imputación
- **Fecha\_Imputacion**--> Fecha de imputación
- **Codigo\_Proyecto** --> Código del proyecto del que se reporta
- **Fase**--> Fase en la que se realiza la imputación
- **Esfuerzo** --> Esfuerzo que se imputa
- **Porcentaje\_Avance\_Fase** --> Porcentaje de avance de la fase



- **Fecha\_Carga** --> Fecha en la que se da de alta el registro en el sistema

➤ **Solicitudes**

```
| Solicitudes | CREATE TABLE `Solicitudes` (  
  `Id_Solicitud` varchar(30) NOT NULL,  
  `Usuario` varchar(15) NOT NULL,  
  `Tipo_Solicitud` varchar(30) NOT NULL,  
  `Desc_Solicitud` varchar(250) NOT NULL,  
  `Fecha_Inicio` date NOT NULL,  
  `Fecha_Fin` date NOT NULL,  
  `Estado` varchar(10) NOT NULL,  
  `Fecha_Carga` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
```

Figura 30 - DLL tabla Solicitudes

Descripción de Campos:

- **Id\_Solicitud** --> Identificador único de solicitud
- **Usuario** --> Login del empleado que realiza la solicitud
- **Tipo\_Solicitud** --> Tipo de solicitud (Vacaciones, Ausencia, etc)
- **Desc\_Solicitud** --> Descripción de la solicitud
- **Fecha\_Inicio** --> Fecha\_Inicio Solicitud
- **Fecha\_Fin** --> Fecha\_Fin\_Solicitud
- **Estado** --> Estado de la solicitud
- **Fecha\_Carga** --> Fecha de última modificación de la solicitud

## 5.4 - Modelo Entidad/Relación de la BBDD

El modelo entidad-relación, es una de las mejores formas de representar la estructura de una base de datos relacional.

Este tipo de diagramas, sirve de gran ayuda a la hora de entender los datos existentes dentro de una base de datos, así como para comprender como se relacionan entre ellos.

### 5.4.1 - Diagrama Entidad/Relación

A continuación, se representa el diagrama entidad relación correspondiente al proyecto de gestión de cartera

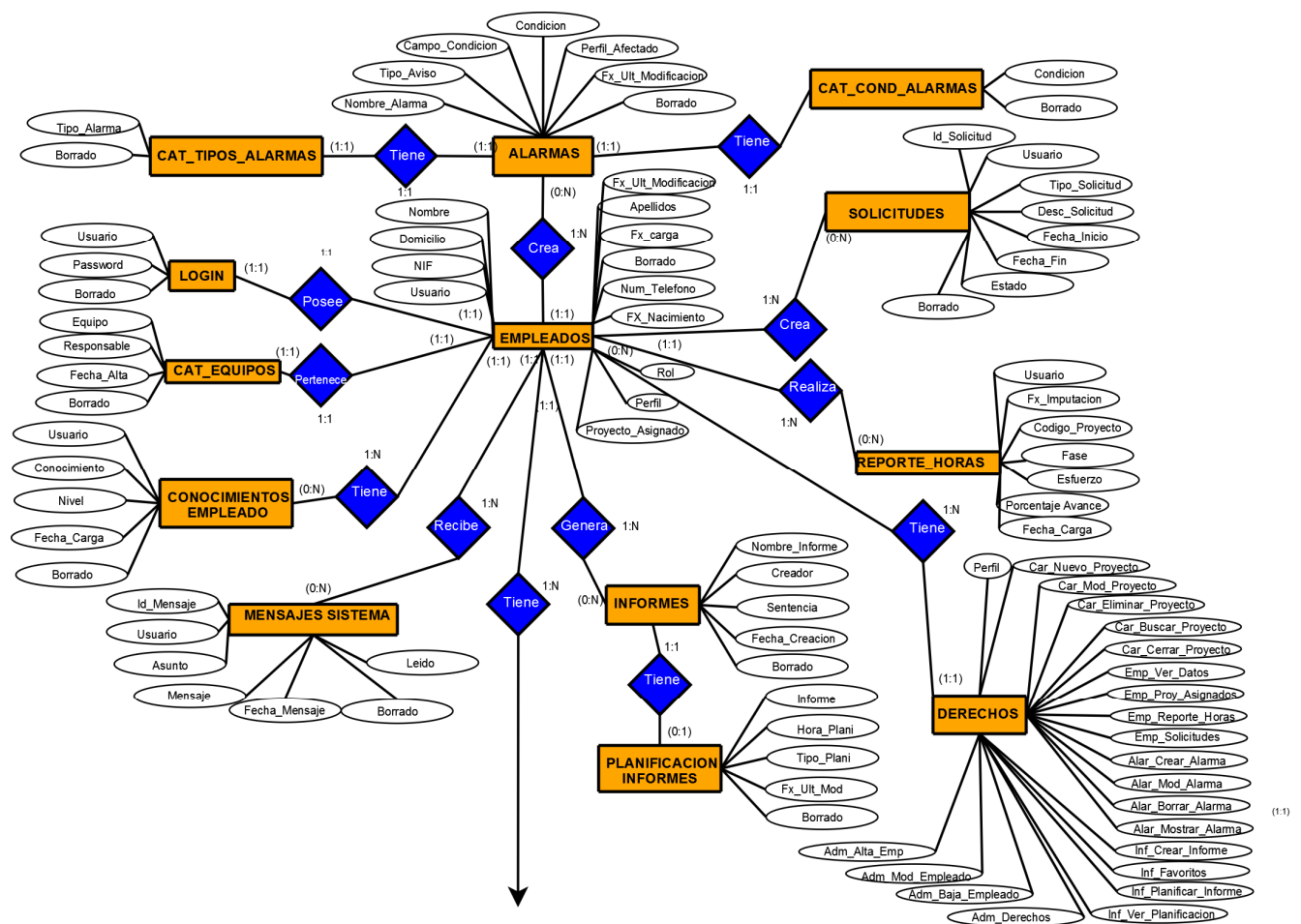


Figura 31 - Grafica Diagrama Entidad Relación

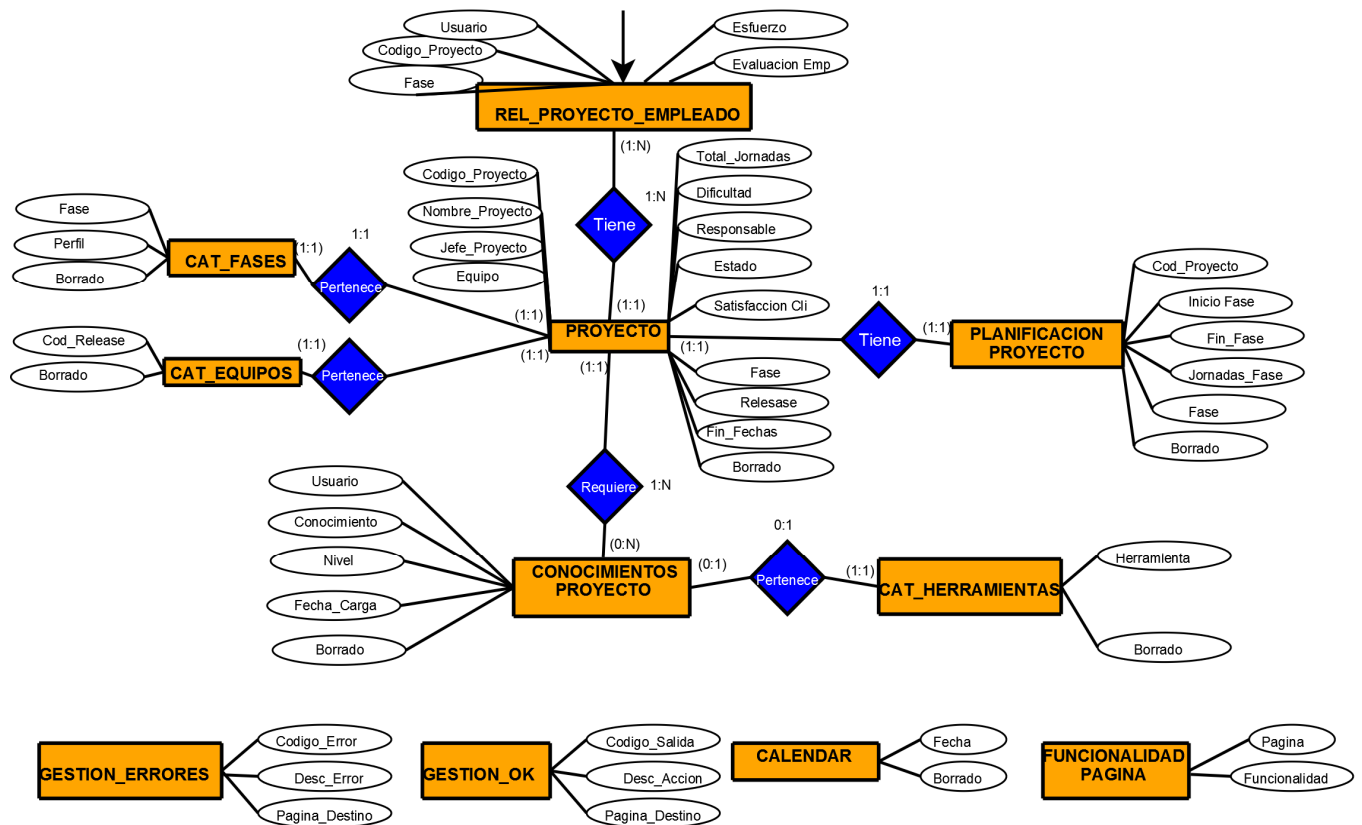


Figura 32 - Grafica Diagrama Entidad Relación (cont.)

#### 5.4.2 - Supuestos Semánticos

- Una fase de un proyecto solamente podrá llevarse a cabo por un único empleado.
- Un informe puede tener como mucho una planificación.
- Una alarma, solo puede aplicar sobre un único perfil de empleado.
- Un proyecto solo puede tener una planificación
- Un empleado tiene siempre un derecho asignado dependiendo de su perfil.
- Un proyecto puede estar en una única fase. No puede haber dos fases o más fases concurrentes.
- Un reporte de horas, solo puede pertenecer a un empleado

## 5.5 - Modelo Relacional de la BBDD

El modelo relacional de una base de datos, se basa en la representación de la información en forma de tablas y relaciones, donde cada fila de la tabla se interpreta como una relación ordenada de valores relacionados entre sí (relacionados entre sí).

### 5.5.1 - Diagrama Relacional

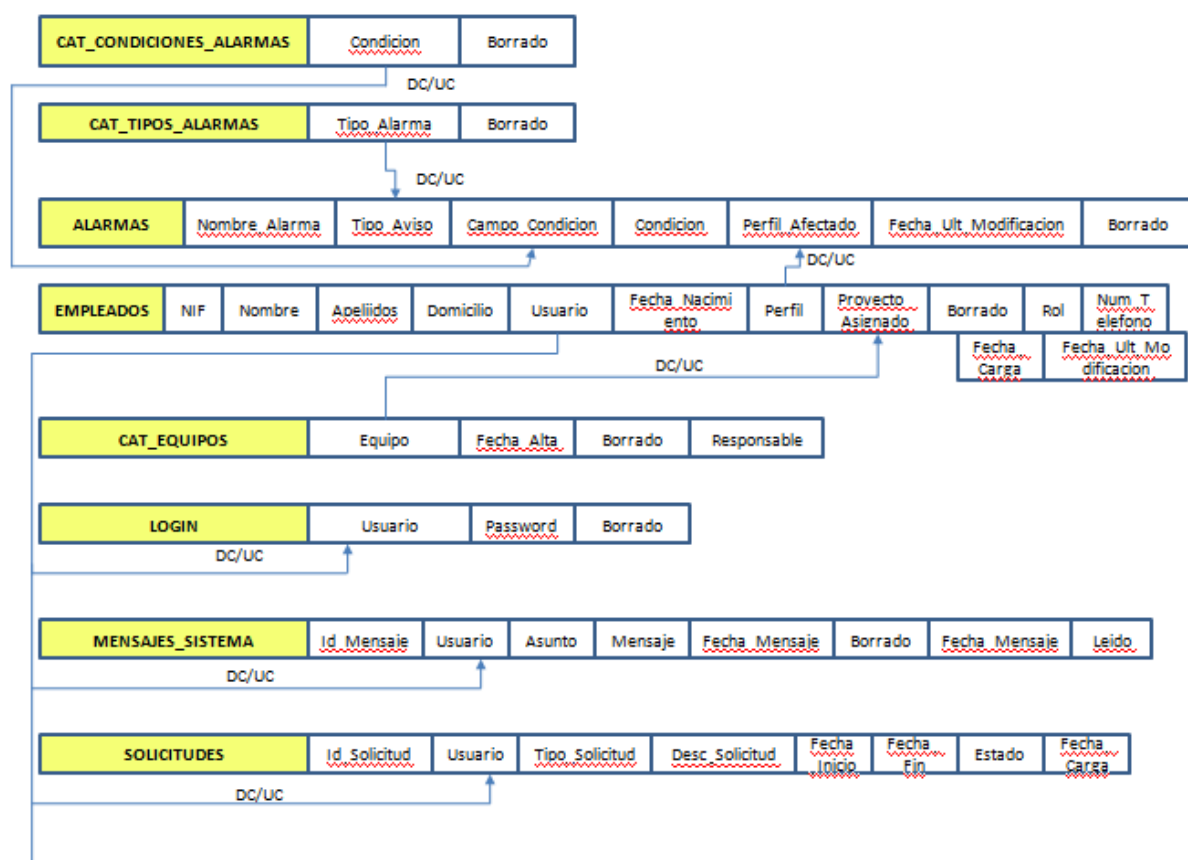


Figura 33 - Diagrama Relacional de Base de Datos

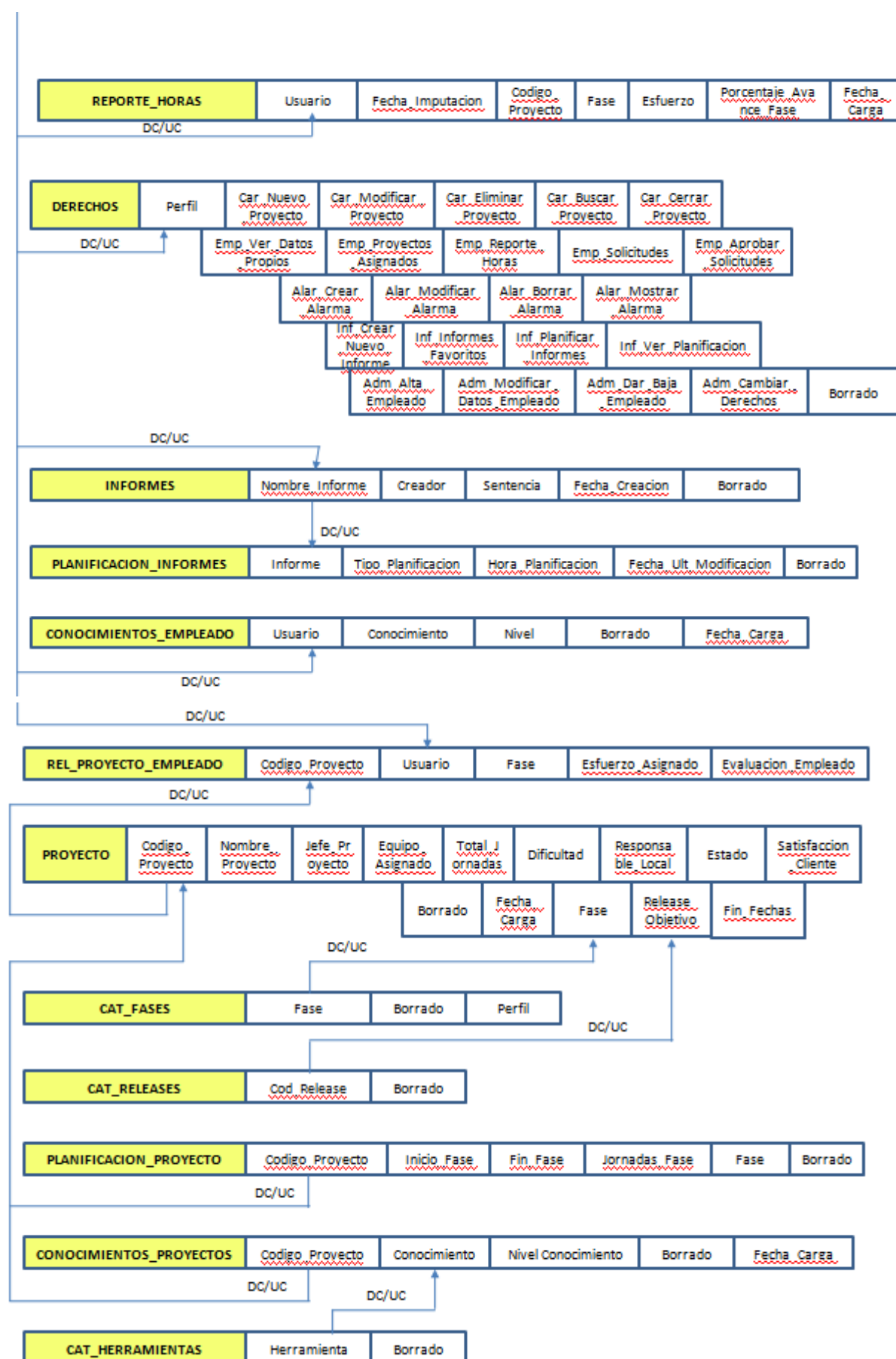


Figura 34 - Diagrama Relacional de Base de Datos (cont.)

Entidades Internas a la funcionalidad del sistema:



Figura 35 - Diagrama Relacional de Base de Datos (cont.2)

### 5.5.2 - Supuestos Semánticos

- En las tabla de planificación proyecto, la fecha de Fin de Fase, siempre tiene que ser mayor o igual que la fecha de Inicio de Fase.
- El campo Borrado de todas las tablas de la base de datos, será 0 cuando el registro sea considerado como activo y 1 cuando se considere como desactivo

## 5.6 - Esquema de BBDD

Dentro de un esquema de base de datos, se define la estructura de la base de datos que será utilizada por la aplicación para almacenar los datos

El esquema que se propone a continuación será el esquema definitivo de la BBDD:

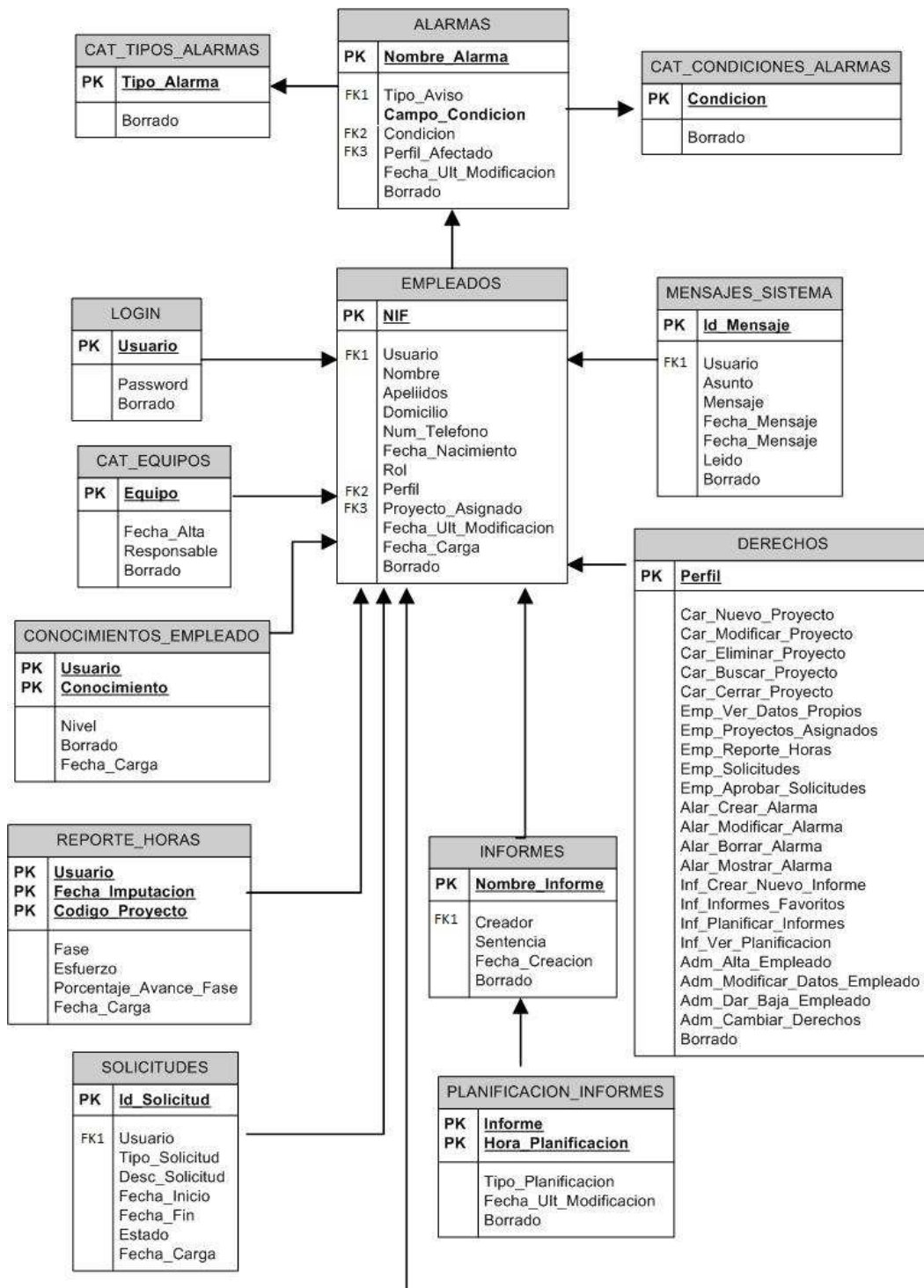


Figura 36 - Esquema de Base de Datos

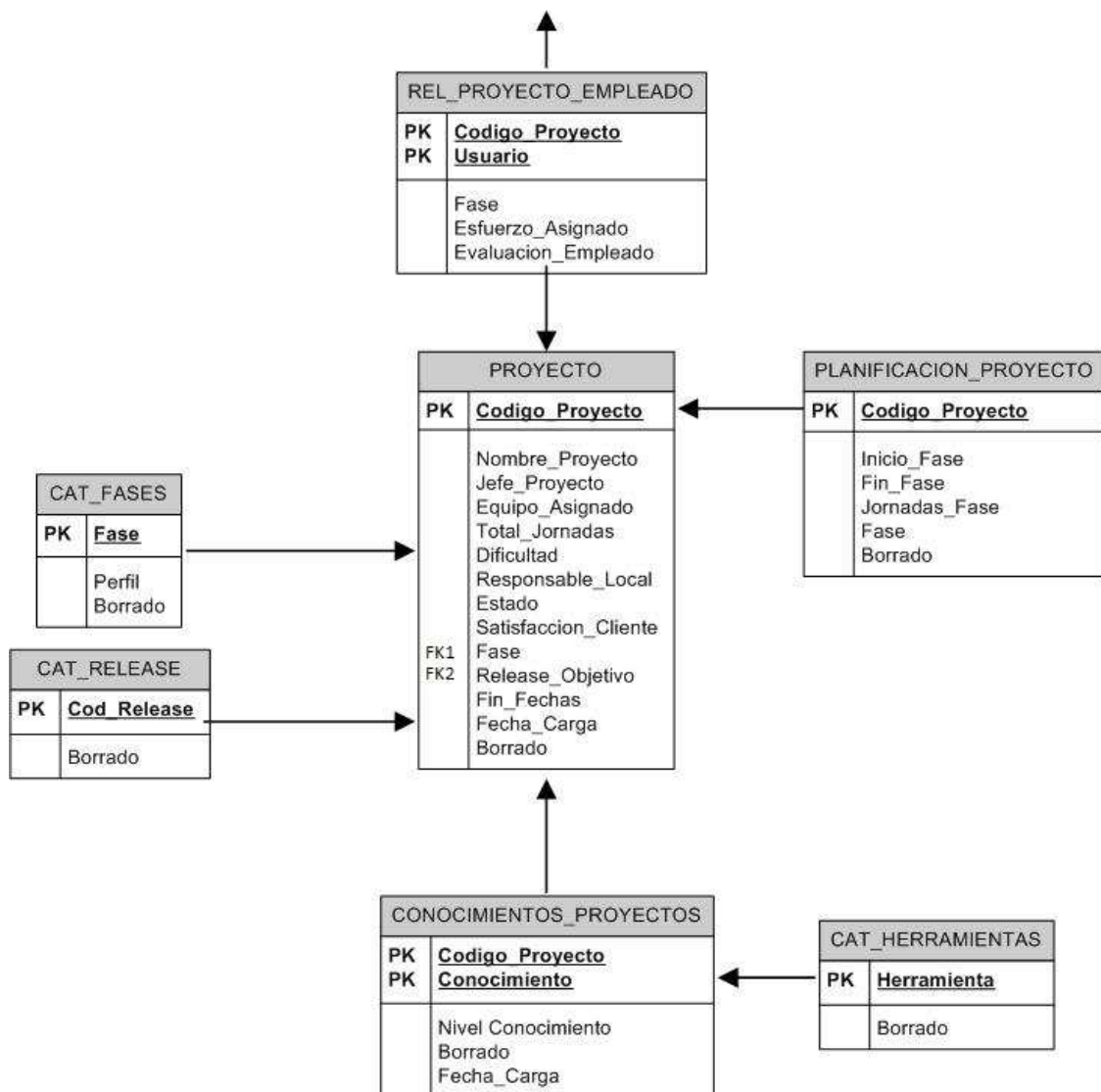


Figura 37 - Esquema de Base de Datos (cont.)

### Esquema de tablas internas a la aplicación:

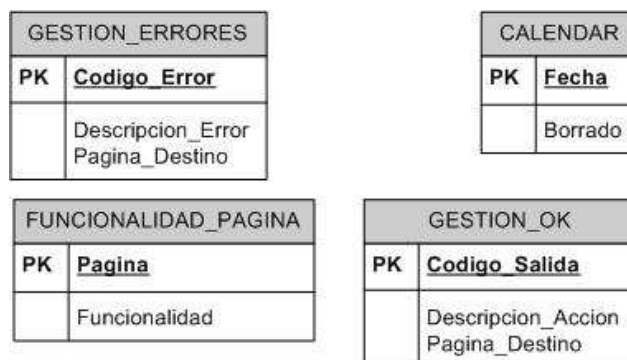


Figura 38 - Esquema de Base de Datos (cont.2)



### 5.6.1 - Cruces BBDD

- Cat\_Tipos\_Alarmas.tipo\_alarma=Alarmas.tipo\_aviso
- Alarmas.campo\_condicion=Cat\_Condiciones.condicion
- Alarmas.perfil\_afectado=Empleados.perfil
- Conocimientos\_Empleados.Conocimiento=Cat\_Herramientas.herramienta
- Conocimientos\_Proyecto.Conocimiento=Cat\_Herramientas.herramienta
- Empleados.perfil=Derechos.perfil
- Empleados.proyecto\_asignado=Cat\_Equipos.equipo
- Empleados.usuario=Conocimientos\_Empleados.usuario
- Empleados.usuario=Informes.creador
- Empleados.usuario=Rel\_Proyecto\_Empleado.usuario
- Empleados.usuario=Solicitudes.usuario
- Empleados.usuario=Reporte\_Horas.usuario
- Informes.nombre\_informe=Planificacion\_Informes.informe
- Login.usuario=Empleados.usuario
- Mensajes\_Sistema.usuario=Empleados.usuario
- Proyecto.codigo\_proyecto=Conocimientos\_Proyecto.codigo\_proyecto
- Proyecto.Codigo\_proyecto=Planificacion\_Proyecto.codigo\_proyecto
- Proyecto.fase=Cat\_Fases.fase
- Proyecto.release\_objetivo=Cat\_Releases.cod\_release
- Rel\_Proyecto\_Empleado.codigo\_proyecto=Proyecto.codigo\_proyecto

## 5.7 - Diseño General del Diagramas de Clases

Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, orientados a objetos. El diagrama de clases de la aplicación gestión de cartera, tanto a alto, como bajo nivel, se mostrará en los siguientes apartados:

### 5.7.1 - Diseño Detallado del Diagrama de Clases

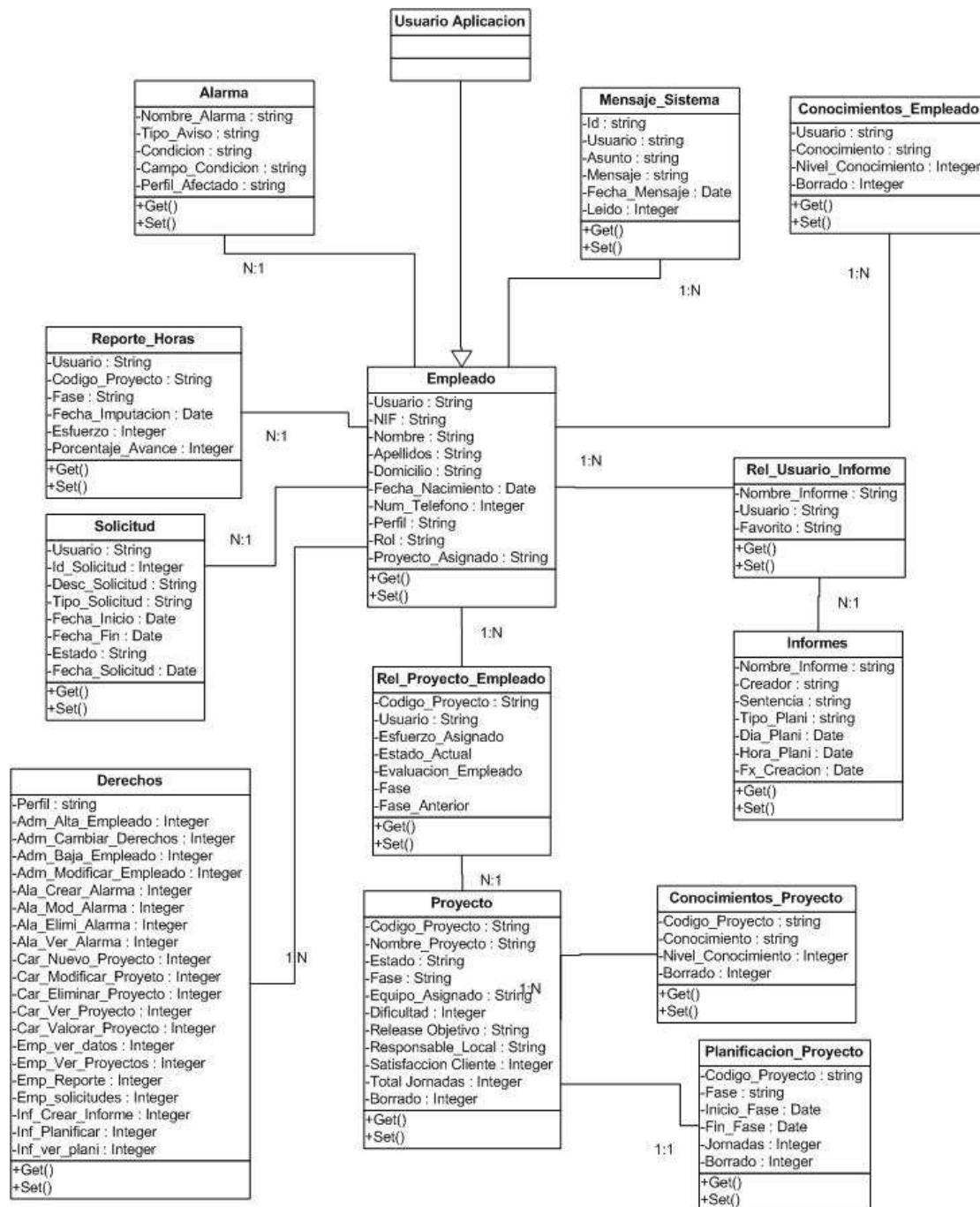


Figura 39 - Diagrama de clases general

## 5.7.2 - Diagrama Arquitectónico de Alto Nivel

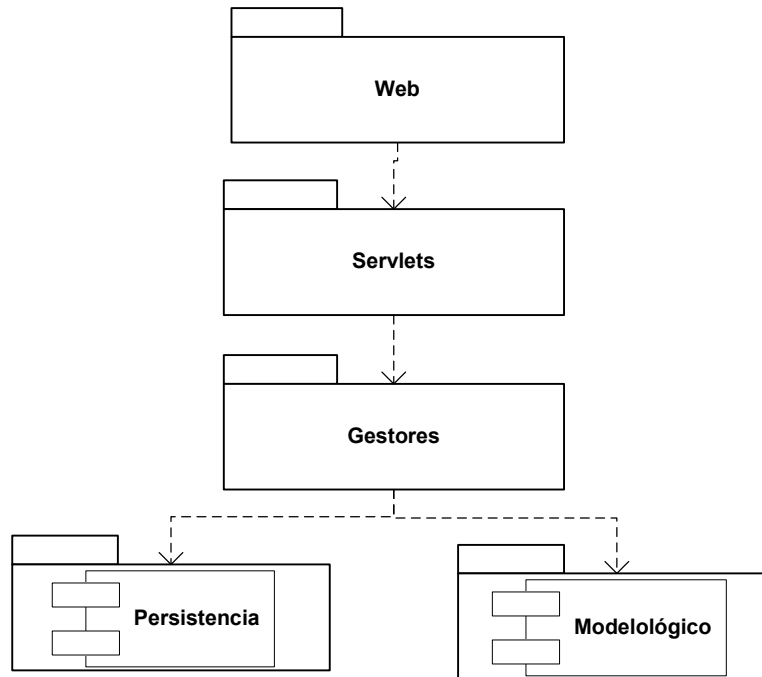


Figura 40 - Diagrama arquitectónico de alto nivel

## 5.7.3 - Diagrama de Clases específico por Paquetes

### 5.7.3.1 - Web vs Servlet

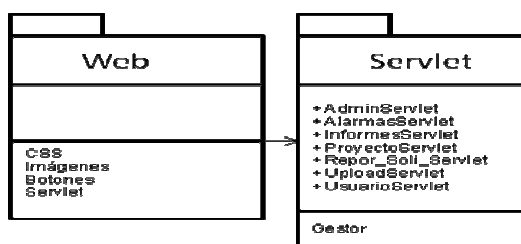


Figura 41 - Diagrama clases Web vs Servlet

### 5.7.3.2 - Servlet vs Gestor

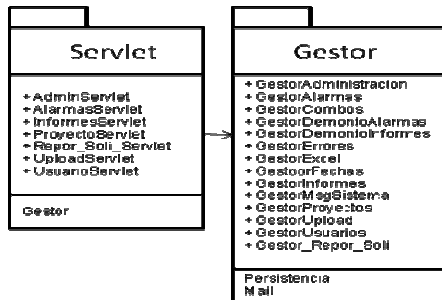


Figura 42 - Diagrama clases Servlet vs Gestor

### 5.7.3.3 - Gestor vs Modelológico

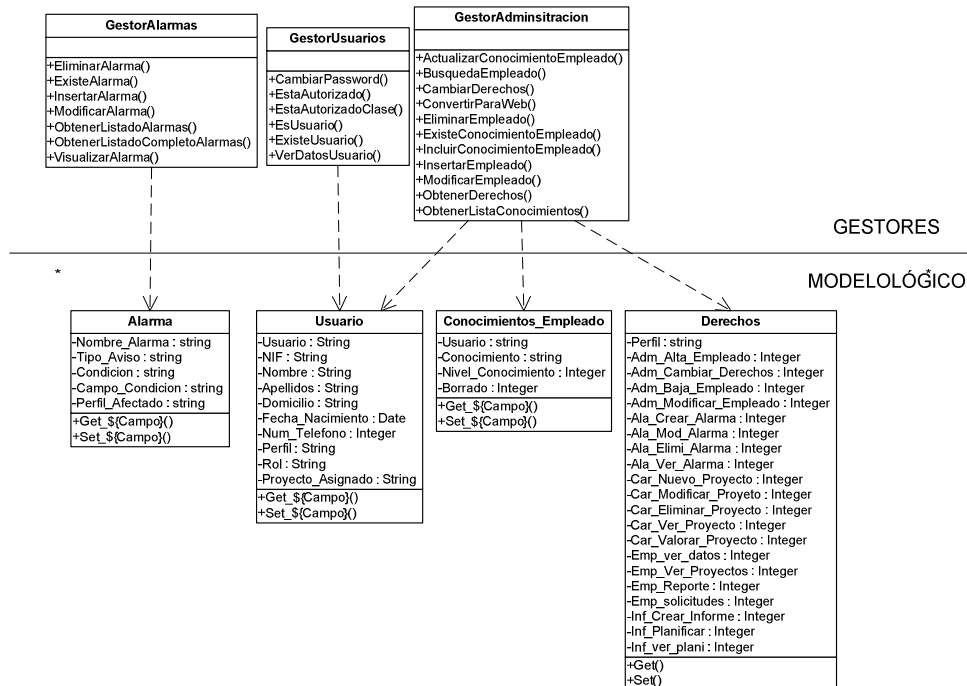


Figura 43 - Diagrama clases Gestor vs Modelológico

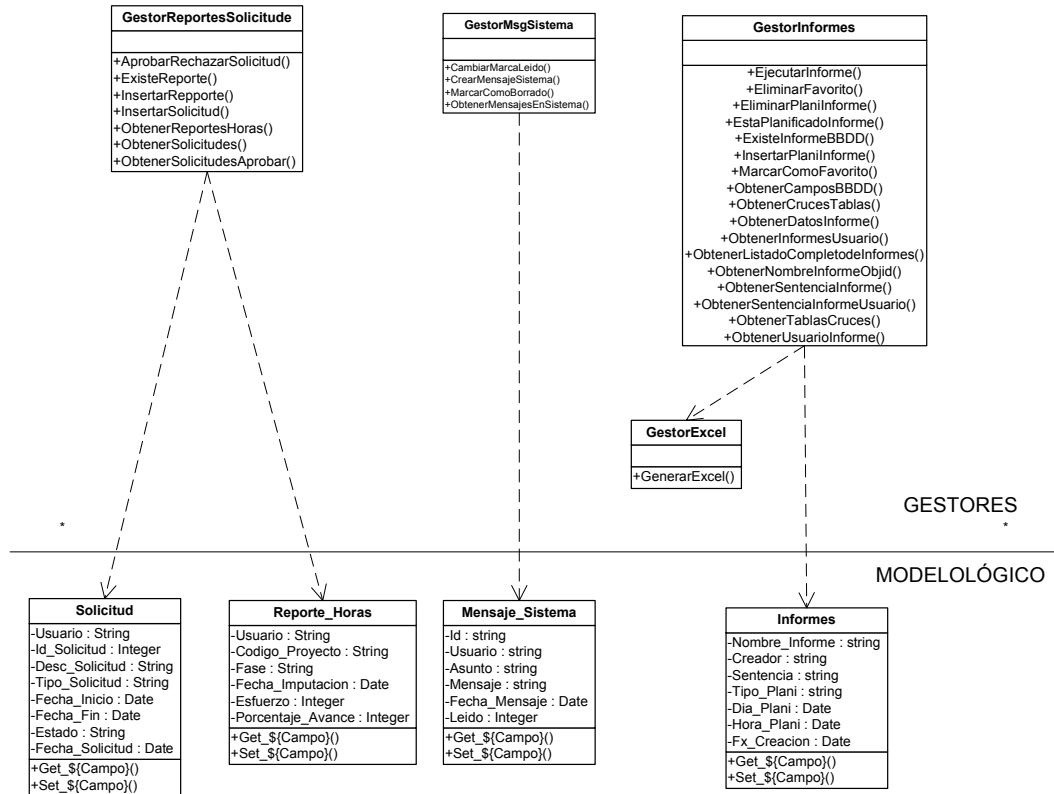


Figura 44 - Diagrama clases Gestor vs Modelológico ( cont.)

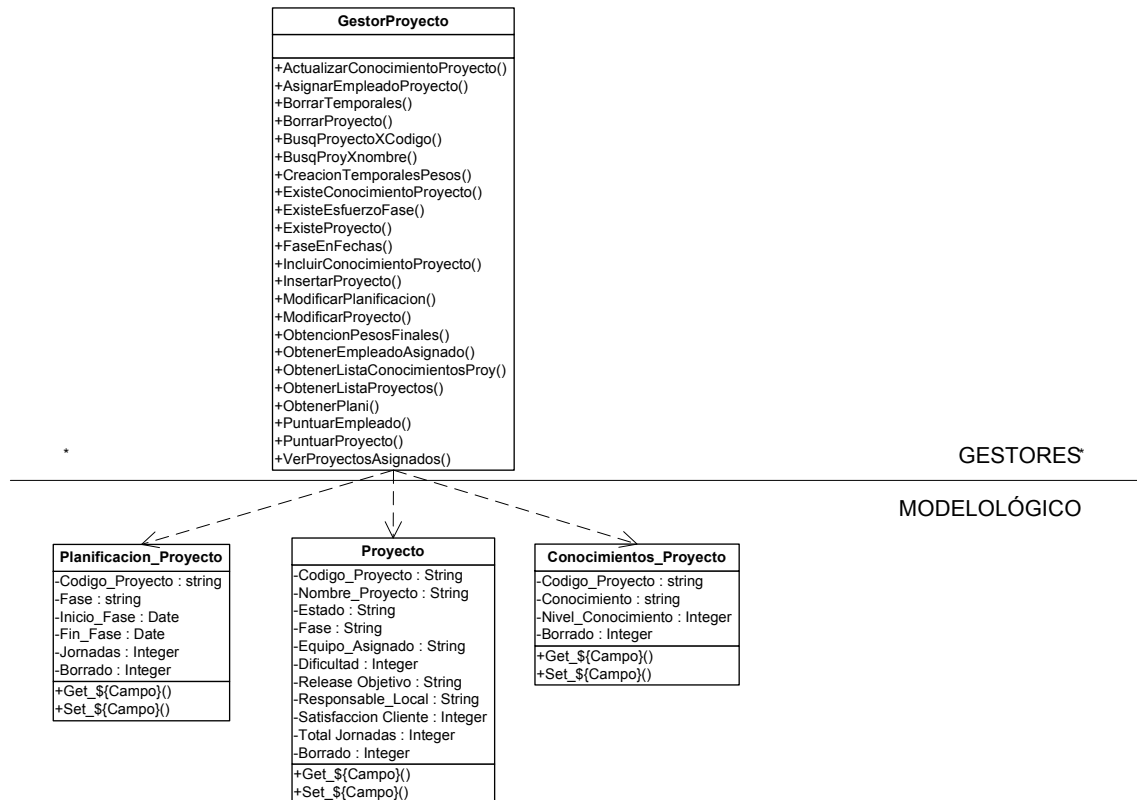


Figura 45 - Diagrama clases Gestor vs Modelológico ( cont.2)

### 5.7.3.5 - Gestor vs Persistencia

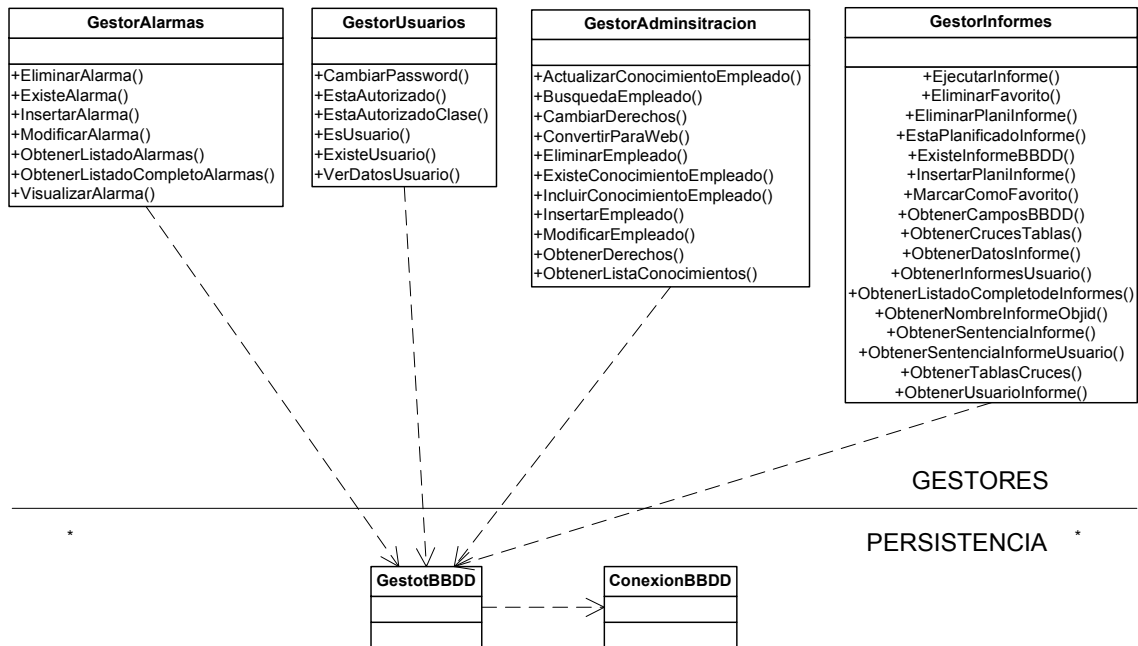


Figura 46 - Diagrama clases Gestor vs Persistencia

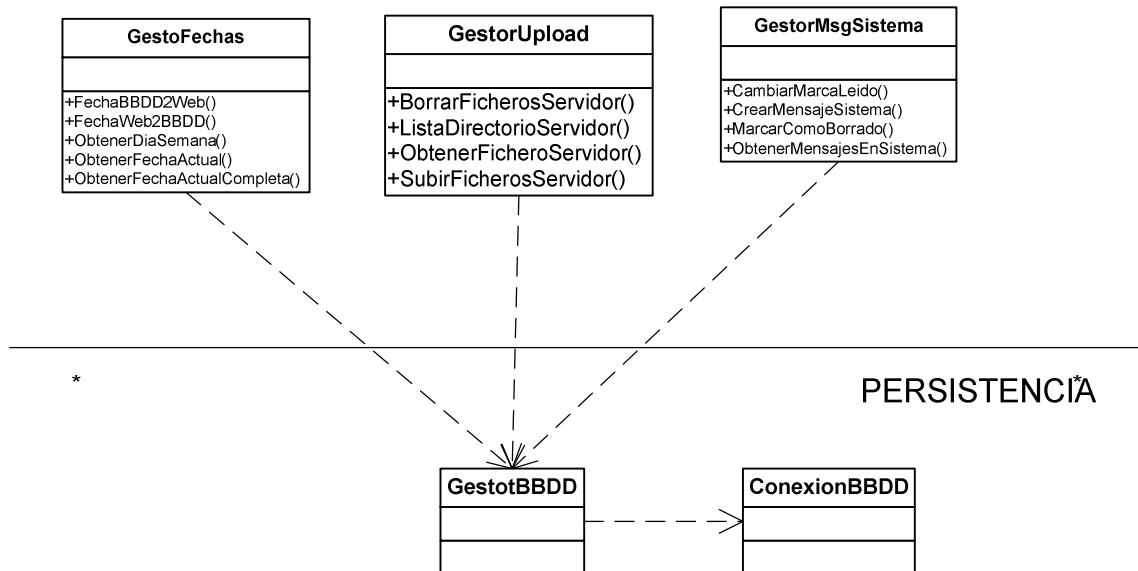


Figura 47 - Diagrama clases Gestor vs Persistencia (cont.)

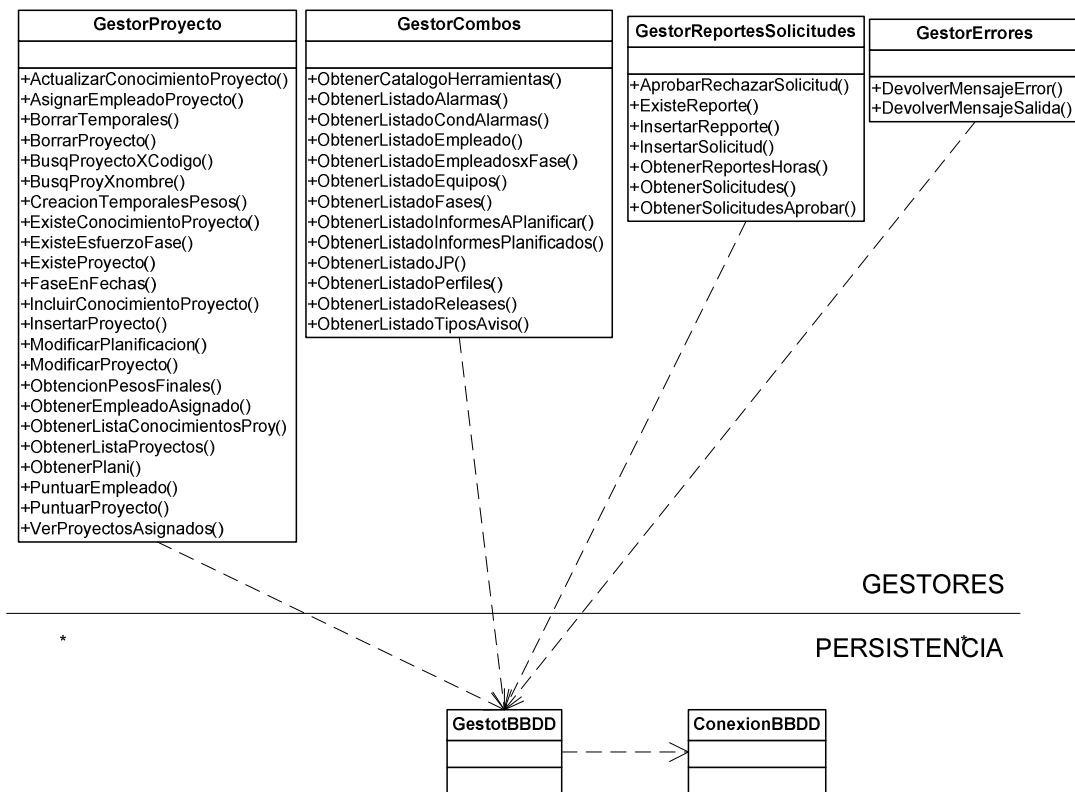


Figura 48 - Diagrama clases Gestor vs Persistencia (cont.2)

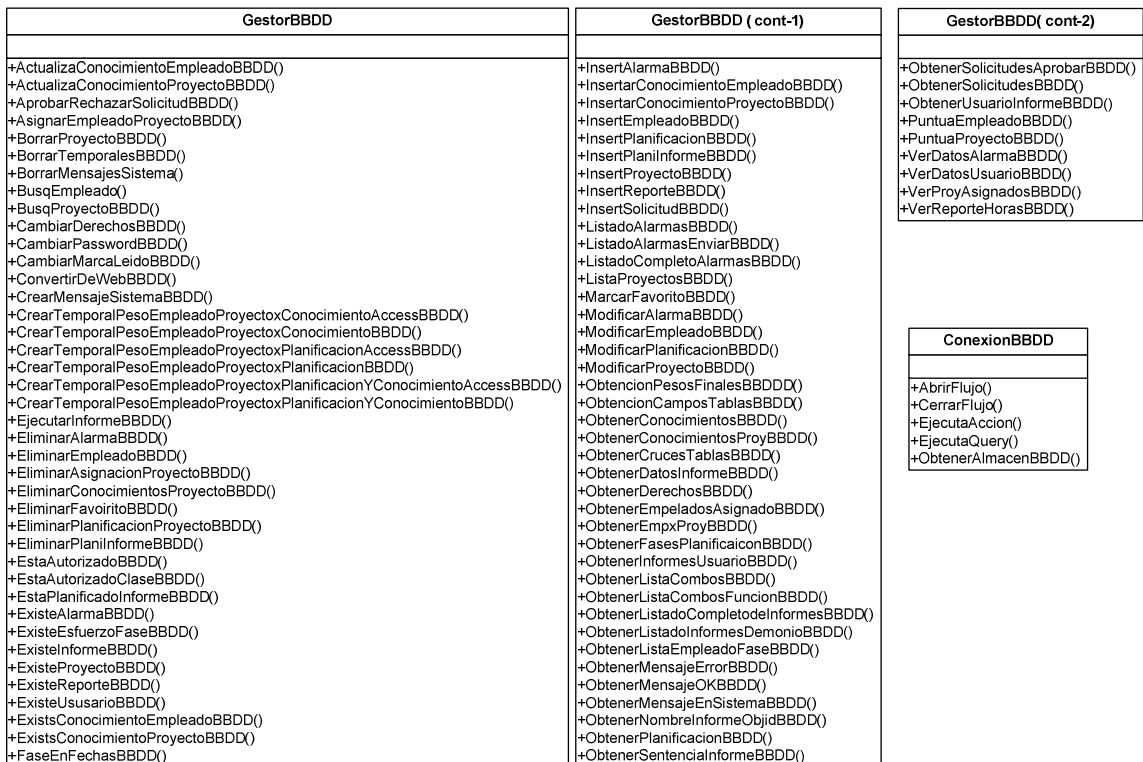


Figura 49 - Diagrama clases Gestor vs Persistencia (cont.3)

### 5.7.3.6 - Otros paquetes

es.uc3m.princon3.mail

Paquete de envío de correo electrónico a servidor externo

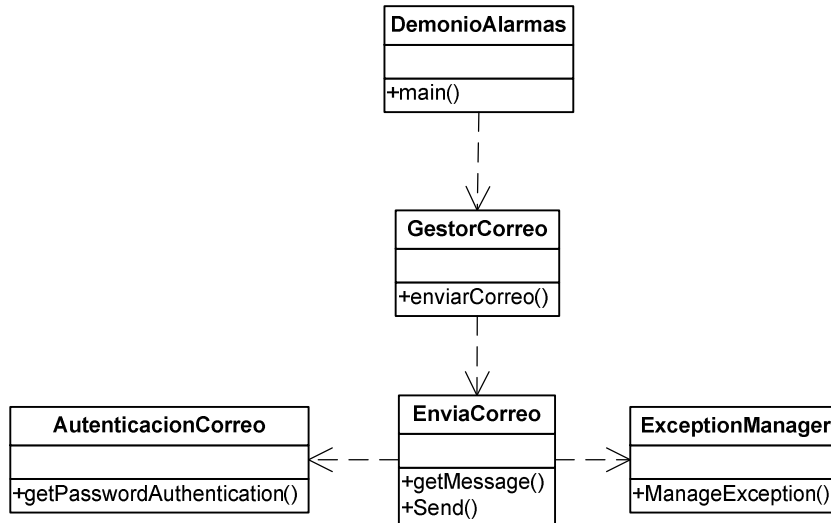


Figura 50 - Diagrama clases paquete mail

es.uc3m.princon3.demonios

Paquete con los procesos demonio que se ejecutarán en segundo plano en el sistema

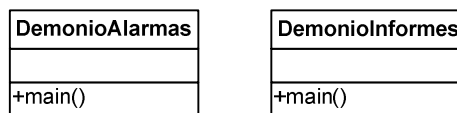


Figura 51 - Diagrama demonios

lanza\_demonio

Paquete con los lanzadores que se utilizarán para poner en marcha los demonios.

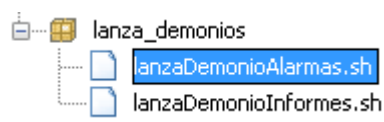


Figura 52 - Lanzadores demonios



## 5.8 - Diseño del proceso de ejecución “Demonios”

Dentro de los requisitos ofrecidos por el usuario, se ha definido una necesidad por la cual es necesario ejecutar una serie de procesos de forma automática diariamente y a determinadas horas del día.

Estas acciones no pueden ser realizadas de forma manual, por lo que se han definido una serie de procesos "demonio", los cuales estén ejecutándose continuamente en segundo plano dentro del sistema, y que reaccionen ante el cumplimiento de las distintas condiciones de ejecución.

Los dos tipos de demonios que han sido definidos son los siguientes:

- **Demonio Alertas** - Comprobará si dentro del sistema el usuario ha definido alguna alerta que se cumpla para el momento de la ejecución, y en caso de existir, se ejecutarán las acciones definidas en la alarma
- **Demonio Informe** - Comprobará si existe algún informe planificado por parte del usuario para ejecutarse a la hora de ejecución del demonio, y en caso de ser así, se obtendrá la sentencia sql almacenada para dicho informe, se ejecutará sobre BBDD y se almacenará el resultado en un documento Excel, dentro del repositorio del usuario que generó y planificó el informe.

A continuación se pasa a detallar, cada uno de los procesos demonio.

### 5.8.1 - Demonio de Alertas

Este demonio se ejecutará 1 vez al día, siempre con horario 00:00, el cual comprobará las alarmas que hay dentro del sistema, comprobarán si estas cumplen las condiciones y en caso afirmativo, ejecutarán la acción determinada para cada una de las alarmas.

Cuando el demonio se ejecuta, realiza los siguientes pasos

- Se comprueba si la hora de ejecución son las 00:00.
- Se obtiene mediante una consulta a BBDD todas aquellas alarmas que cumplen las condiciones de cara a ser ejecutadas.
- Por cada una de las alarmas que se hayan cumplido se ejecuta la acción que tengan definidas cada una ( mail, mensaje en sistema, ...)
- Finalmente se calcula el tiempo que queda para que se den las 00:00 del día siguiente, y se "duerme" el proceso hasta que pase el tiempo calculado.

5.8.1.1 - Diagrama de flujo del proceso Demonio de Alertas

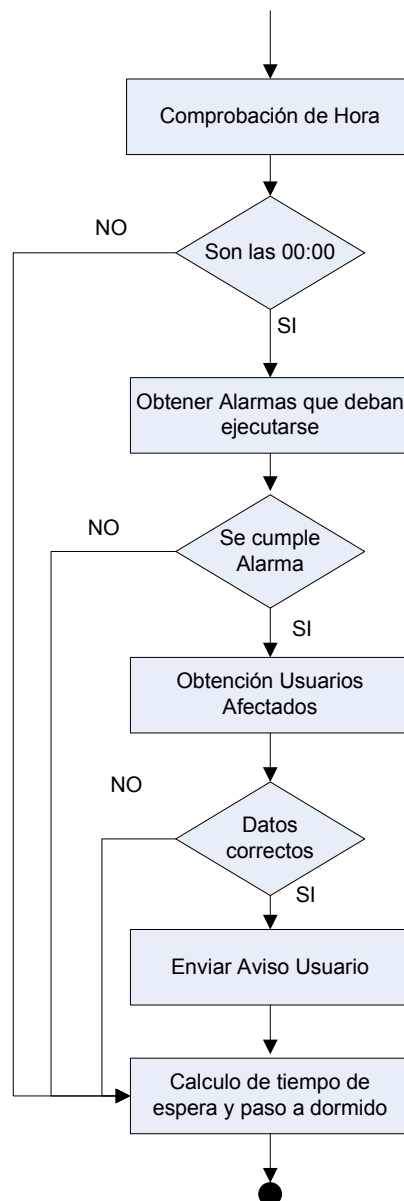


Figura 53 - Diagrama de Flujo de Demonio de Alarmas

### 5.8.2 - Demonio de Informes

Este demonio se ejecutará diariamente cada 5 minutos, que es el margen que se ha establecido dentro de la aplicación para que un usuario pueda ejecutar un informe.

El proceso demonio comprobará los informes que hay planificados dentro del sistema, y en caso de que los campos día ejecución y hora ejecución se cumplan, se procederá a ejecutar el informe seleccionado y depositarlo dentro del repositorio del usuario que generó dicho informe.

Cuando el demonio se ejecuta, realiza los siguientes pasos

- Se comprueba si el minuto actual de la hora del sistema es 00, 05, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55.
- Se obtiene según el día y la hora de ejecución ,los informes por cada usuario que es necesario ejecutar.
- Por cada uno de los informe que haya que seleccionados, se obtendrá la sentencia sql de dicho informe y se ejecutará contra BBDD.
- El resultado del informe será almacenado en un Excel y depositado en el repositorio del usuario.
- Finalmente se calcula el tiempo que queda para la siguiente ejecución, y se "duerme" el proceso hasta que pase el tiempo calculado.

5.8.2.1 - Diagrama de flujo del proceso Demonio de Informes

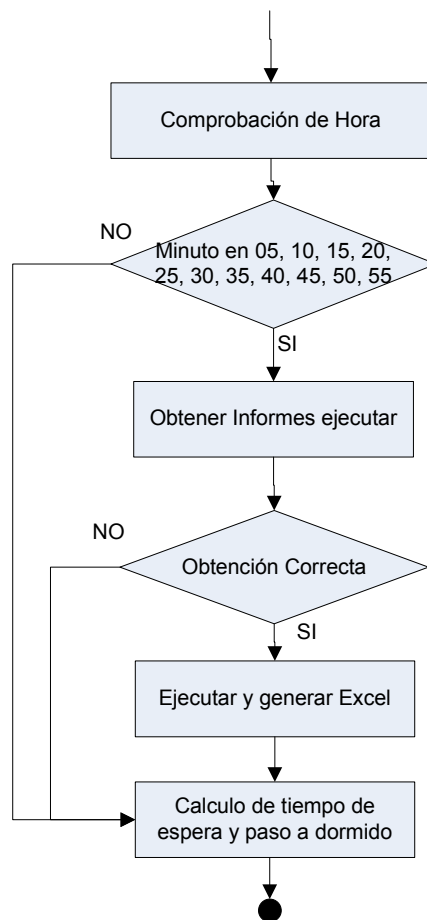


Figura 54 - Diagrama de Flujo de Demonio de Informes

## 5.9 - Diagramas de secuencia

### 5.9.1 - Introducción

Los diagramas de secuencia tratan de mostrar la interacción de los distintos elementos en un sistema a través del tiempo, desde la solicitud inicial del actor principal, hasta obtener un resultado.

Debido a la gran funcionalidad que posee esta aplicación, y el gran número de casos de uso que la misma tiene, se ha optado por elegir una serie de casos de uso como representativos, para mostrar la secuencia que los distintos casos de uso generan dentro del sistema.

Los casos de uso seleccionados han sido los siguientes:

- Creación de nuevo proyecto dentro del sistema
- Creación de nueva alarma
- Generación de nuevo informe
- Alta de nuevo empleado.
- Autoasignación de empleados a proyecto

### 5.9.2 - Diagrama secuencia de creación de nuevo proyecto

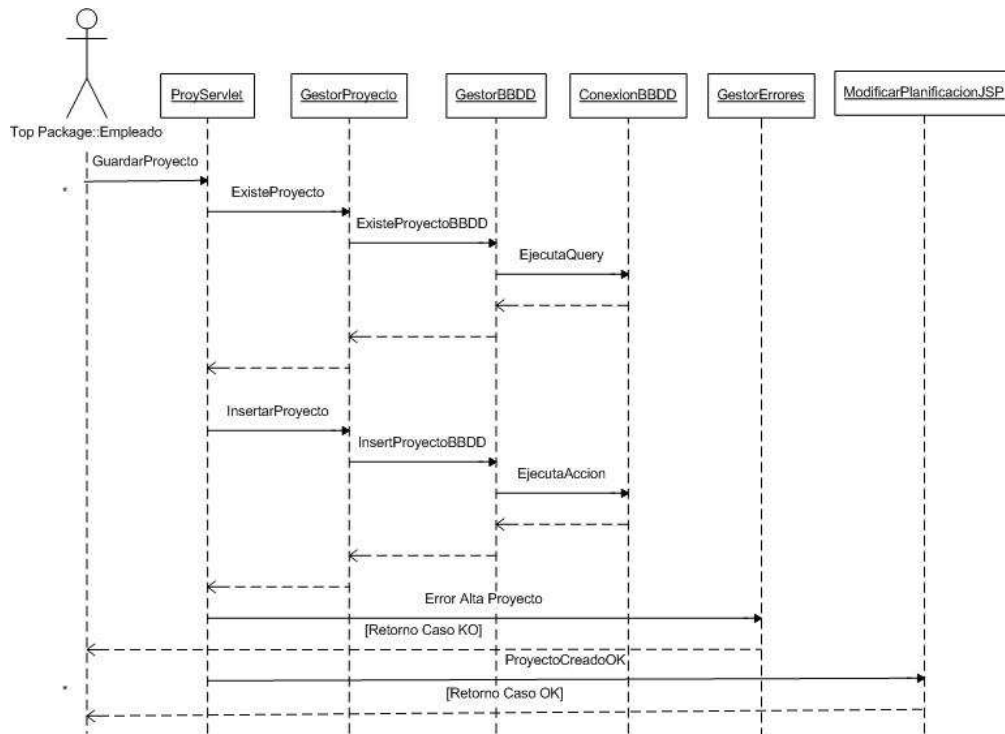


Figura 55 - Diagrama Secuencia creación de proyecto

### 5.9.4 - Diagrama secuencia creación de nueva alarma

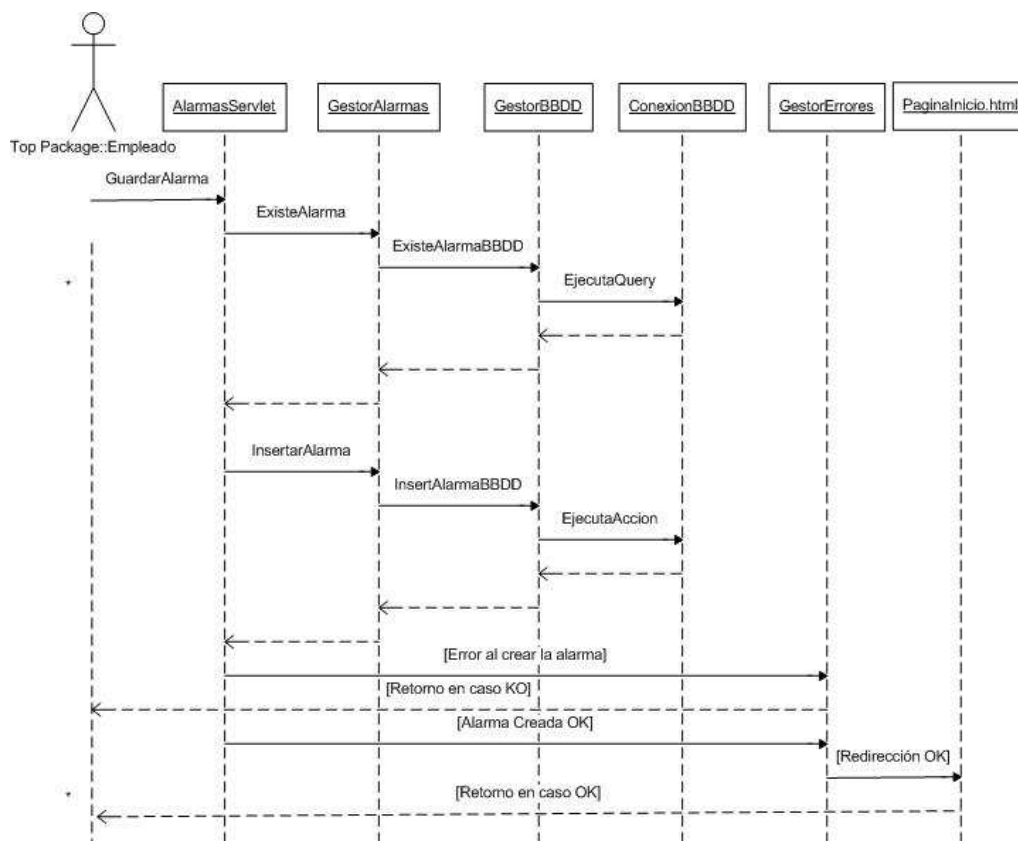


Figura 56 - Diagrama Secuencia creación de alarma

## 5.9.5 - Diagrama secuencia de generación de nuevo informe

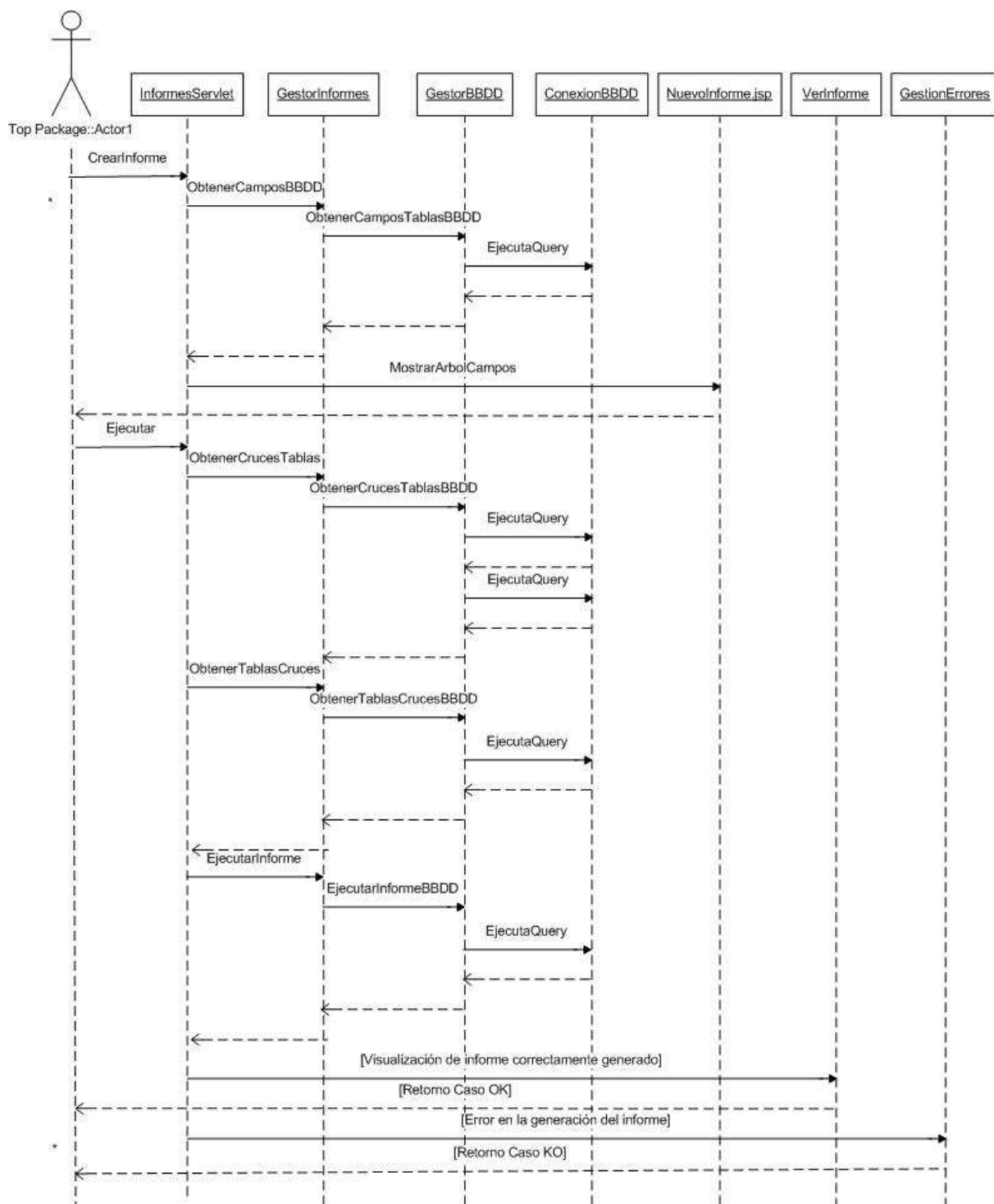


Figura 57 - Diagrama Secuencia creación de nuevo informe



### 5.9.6 - Diagrama secuencia de alta de nuevo empleado.

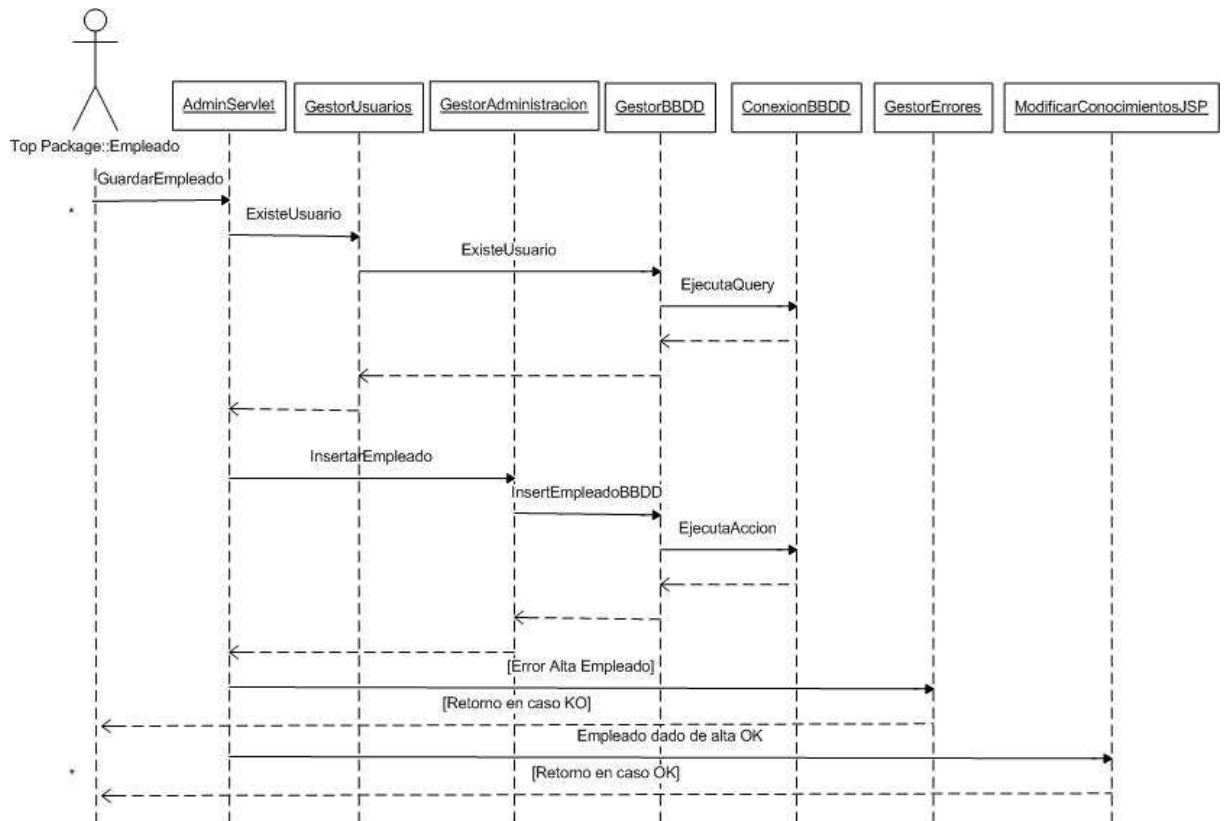


Figura 58 - Diagrama Secuencia creación de nuevo empleado

## 5.9.7 - Diagrama secuencia de autoasignación de empleados a proyecto

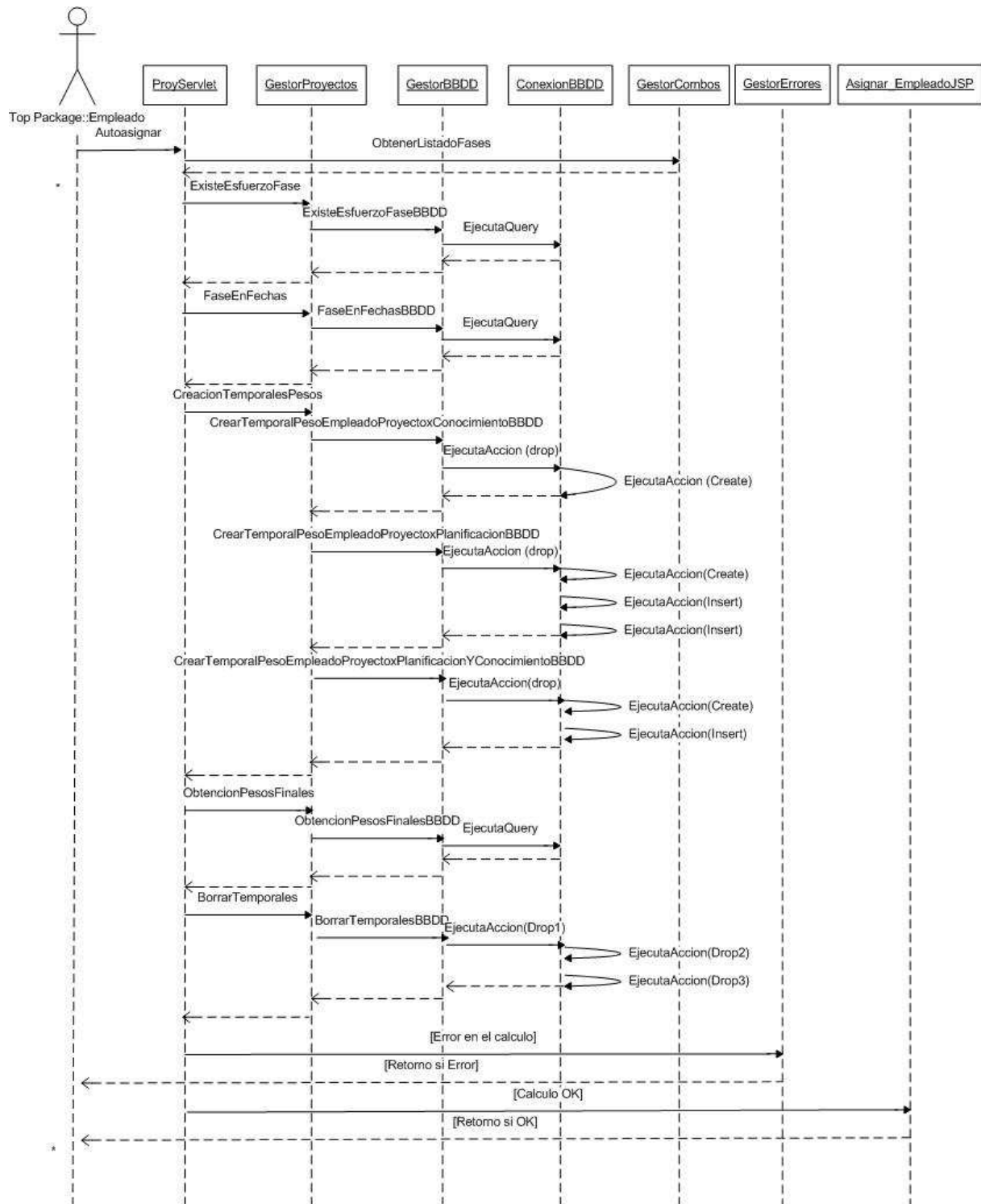


Figura 59 - Diagrama Secuencia autoasignación de proyecto

## 6 - Codificación

Dentro del apartado de codificación del proyecto, veremos por un lado la estructura que se ha seguido dentro del desarrollo del proyecto, y por otro lado las diferentes normas de codificación que se han tenido en cuenta de cara a que el desarrollo realizado por los distintos programadores, y el entendimiento del programas sea sencillo y homogéneo.

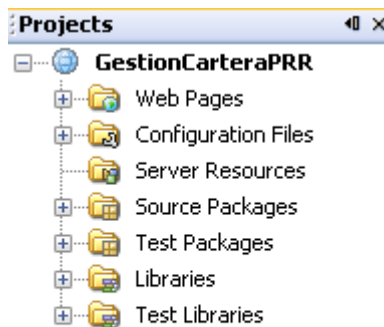
Finalmente, se detallarán los distintas librerías externas que han sido utilizadas dentro del proyecto, así como su funcionalidad.

## 6.1 - Estructura de proyecto Java

El proyecto de Gestión de Cartera ha sido implementado mediante la herramienta NetBeans 5.5, la cual contiene internamente dos tipos de estructuras, una la estructura de proyecto, y otra la estructura de carpetas.

De esta manera se detalla a continuación el detalle del contenido de cada una de ellas:

### Estructura de proyecto:



---

Figura 60 - Estructura de proyecto generada

- **web pages** --> Contiene la información de las página web html y jsp, que son ejecutadas dentro de la aplicación, así como los archivos que son necesarios para que estas sean ejecutadas, como son librerías, imágenes iconos, etc.
- **configuration files** - > Contiene los archivo de configuración de la aplicación
- **server resources** -> N/A para este proyecto
- **source packages** -> Contiene todo el código fuente .java y .sh de la aplicación
- **test packages** --> N/A para este proyecto
- **libraries** --> **Contiene todas la librerías y conectores que son utilizados dentro de los archivos java de la aplicación.**
- **test libraries** --> N/A para este proyecto

Estructura de carpetas:

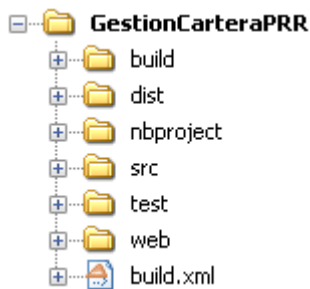
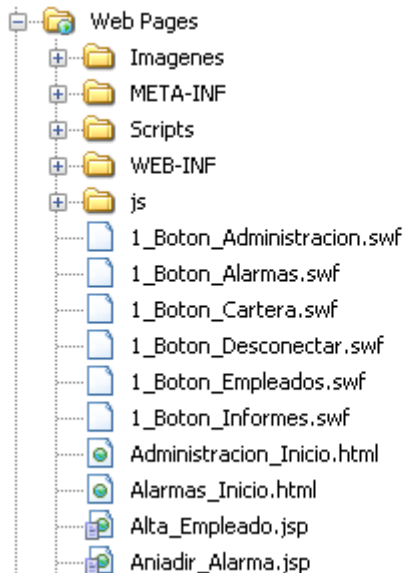


Figura 61 - Estructura de carpetas de proyecto generado

- **build** --> Contiene las clases compiladas (.class)
- **dist** --> dentro de esta carpeta se crea un archivo comprimido con la extensión ".war" que contiene todos los archivos necesarios para distribuir la aplicación generada y que pueda ser ejecutada.
- **src** --> Almacena los archivos con el código fuente (.java) que ha sido escrito o que forman parte de la aplicación.
- **web** --> Contiene todas las páginas .html, .jsp, así como la información necesaria para que estas puedan ejecutarse correctamente, como imágenes, botones, etc.

### 6.1.1 - Estructura de páginas Web

Dentro de la parte web de la aplicación, podemos encontrar la siguiente estructura de carpetas:



**Figura 62** - Estructura de carpetas de parte WEB

Dentro de la propia carpeta de Web Pages, podemos encontrar por un lado los distintos botones del menú principal creados con "Shock Wave Flash Player" y por otro lado las distintas página de tipo html, y jsp

Las páginas de tipo html, son aquellas cuya interacción con la parte java de la aplicación únicamente se produce a modo entrada en la llamada al html, y salida, en la salida del html, sin ningún tipo de código Java en su interior

Por otro lado, las páginas jsp, son aquellas en las que internamente al código se ha incluido código propiamente Java, para o bien realizar alguna acción interna a la web , o bien recuperar algún tipo de dato en tiempo de ejecución del jsp desde la parte java de la aplicación

Dentro de la carpeta js, tendremos las distintas librerías utilizadas desde las páginas web, las cuales se detallan dentro del apartado 5.3.2 de este mismo documento. .

### 6.1.2 - Estructura de paquetes

El proyecto java se encuentra dividido en paquetes siguiendo la siguiente metodología:

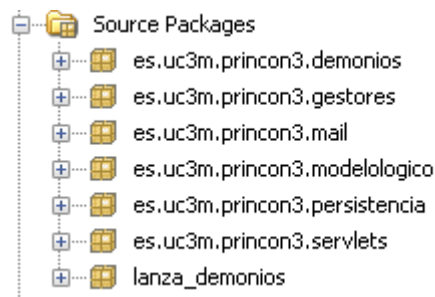


Figura 63 - Detalle paquetes creados para el proyecto

Cada uno de los paquetes contiene información diferenciada, y que ante nuevos desarrollos ha de ser tenida en cuenta:

#### Demonios

El paquete "demonios" contiene la información de los archivos que son ejecutados en segundo plano dentro del servidor a modo de "demonios", y que estarán corriendo constantemente con la intención de ir comprobando las distintas condiciones y ejecutarse de forma automática cuando estas sean cumplidas, sin intervención humana.

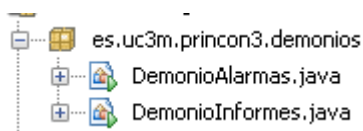


Figura 64 - Detalle demonios creados para el proyecto

## Gestores

El paquete "Gestores", contiene todas las clases con las cuales contiene la funciones necesarias de gestión de las distintas funcionalidades del sistema

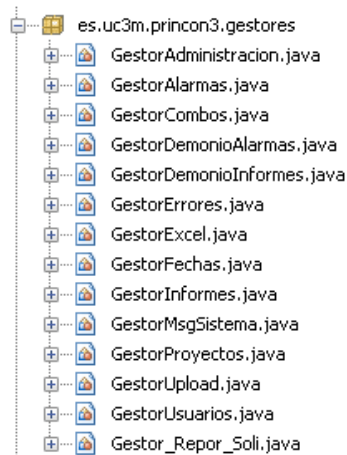


Figura 65 - Detalle gestores creados para el proyecto

## Mail

El paquete "Mail", contiene todas las clases que son necesarias para que pueda llevarse a cabo la funcionalidad de enviar correos electrónicos a servidores de correo externo.

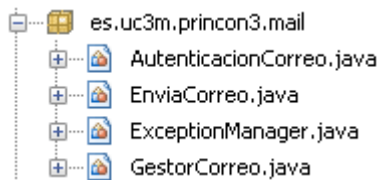


Figura 66 - Detalle paquete mail



### Modelológico

El paquete modelo lógico contiene una clase por cada uno de los modelos a los que se les define atributos y métodos comunes

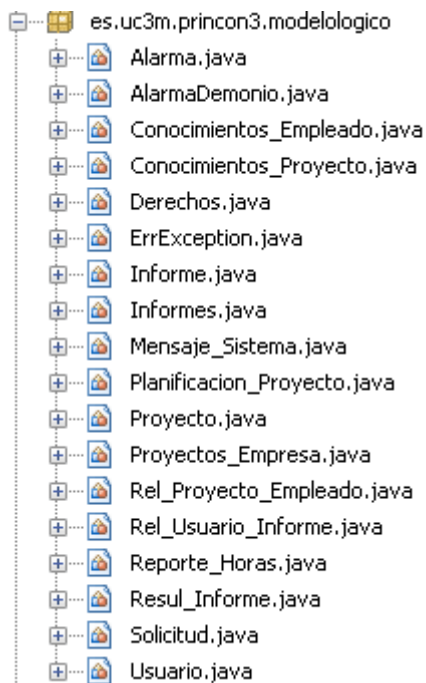


Figura 67 - Detalle paquete modelológico generado en el proyecto

### Persistencia

El paquete "persistencia" define todas las clases tanto de conexión a BBDD, como todas las funciones que realizan algún tipo de conexión a la misma.



Figura 68 - Detalle paquete persistencia generado en el proyecto

## Servlets

El paquete "Servlets" contiene todas aquellas clases que son utilizadas a modo pasarela entre la web y la aplicación java, recibiendo peticiones desde el navegador web, procesándolas y devolviéndole una respuesta.

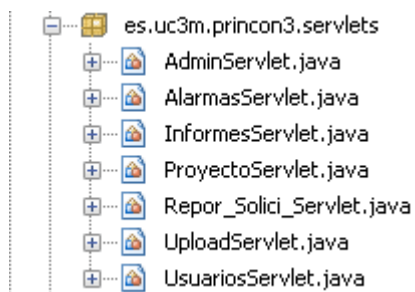


Figura 69- Detalle de servlets generados en el proyecto

## lanza\_demonios

El paquete "lanza\_demonios", contiene los .sh que será necesario ejecutar dentro del servidor, para que los procesos comiencen a ejecutarse en segundo plano.

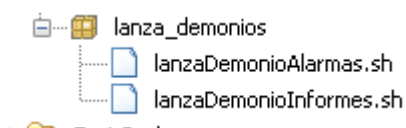


Figura 70 - Lanzadores de los demonios del alarmas e informes

## 6.2 - Normas y consejos de codificación

En todo proyecto de desarrollo informático es de suma importancia establecer y trabajar con una serie de normas de codificación claras y homogéneas de se puedan cumplir una serie de objetivos:


- Todo el equipo de programación tenga claro que nomenclaturas ha de seguir a la hora de codificar
- Facilita el entendimiento del código
- Una vez finalizada la aplicación, cualquier persona ajena al proyecto tenga la capacidad de poder mantenerlo.

Con la aplicación de las normas de codificación no se aprecia una mejora funcional, de rendimiento, o de la eficacia, pero son muy necesarias para facilitar la legibilidad del código escrito en lenguaje JAVA, y servirá como referencia para futuros desarrollos y además aumenta la eficiencia del desarrollo al seguir un proceso de codificación establecido de antemano.

Dentro de este proyecto las reglas básicas de cumplimiento serán las siguientes:

- Código organizado tal y como marca la estructura del programa indicada dentro del apartado 5.1

- Todo Servlet finaliza con el nombre "Servlet.java

 ProyectoServlet.java

- Todo gestor comenzará con el nombre Gestor\*.java

 GestorProyectos.java

- Todo demonio comenzará con la palabra Demonio\*.java

 DemonioAlarmas.java

- Todas las tablas de BBDD tendrán la primera letra de cada palabra en mayúsculas

**Rel\_Proyecto\_Empleado**

- Todo paquete, clase, método ha de estar comentado siguiendo los estándares de javadoc, indicando descripción parámetros de entrada, de salida, ....

```
//-----
} /**
 * <p>FaseEnFechas -Comprueba si una fase de un proyecto, se encuentra dentro de
 * @param cod_proyecto codigo del proyecto del que se quiere comprobar
 * @param Fase fase de la que se quiere comprobar
 * @return true si la fase esta dentro de fechas, y false, si no lo está
```

- Toda sentencia compleja ha de ser documentada sobre el propio código
- Toda sentencia con lógica sencilla y visible, no será documentada.
- Los nombres de las clases y métodos deben de ser cortos y concisos.
- Los nombres de las clases, paquetes, métodos, etc, irán siempre en minúsculas, con la primera letra de cada palabra en mayúsculas
- No se utilizarán separadores de campos entre las palabras, a la hora de nombrar a la clases, paquetes, métodos etc.
- Todo el código del documento deberá estar correctamente tabulado, para mayor legibilidad del programador o lector que quiera leer el código.

- Toda página web creada para el proyecto tendrá la siguiente estructura básica de página, en la que únicamente se verá modificado el formato del cuadro interno con la funcionalidad propia de cada página. Las únicas excepciones serán las páginas de OK, Error e Inicio.

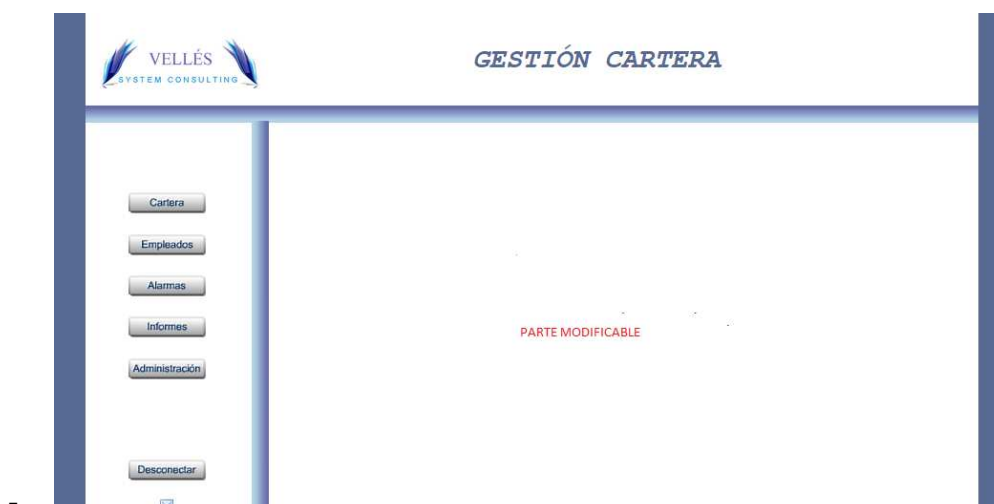


Figura 71 - Estructura principal de la parte web de la aplicación

## 6.3 - Librerías externas utilizadas dentro de la aplicación

Dentro de las librerías que son utilizadas dentro de la aplicación, vamos a diferenciar entre las librerías que son utilizadas para la visualización de datos desde las páginas html y jsp y las que son utilizadas dentro de los propios procesos java.

### 6.3.1- Librerías utilizadas dentro de los procesos Java

Estas librerías son aquellas que son utilizadas dentro de los Servlet, y Gestores de cara a otorgar funcionalidad previamente implementada al proyecto, sin necesidad de tener que codificar todos los métodos nuevamente.

- mysql-connector-java-5.0.8-bin.jar - Librería utilizada para la conexión entre la aplicación y la base de datos Mysql.
- mail.jar y activation.jar - Librerías utilizadas para la autenticación y envío de correos electrónicos a servidores externos desde la aplicación.
- poi-3.10-FINAL-20140208.jar - Librería utilizada para la exportación de datos desde la aplicación a documentos de tipo "Microsoft Excel".
- json-rpc-1.0.jar - Librería para el tratamiento de información en formato JSON

### 6.3.2 - Librerías utilizadas en parte web de la aplicación.

Estas librerías son aquellas que son utilizadas dentro de la parte web de la aplicación, en las páginas de tipo html y jsp, y cuya función se basa en la visualización de datos en un formato determinado, de cara por un lado a otorgar una interfaz más elegante para el usuario, y por otro lado, para ser capaces de realizar acciones que con código html, jsp nos sería muy complicado.

- Dtree --> Utilizada para la visualización de los datos en forma de árbol.

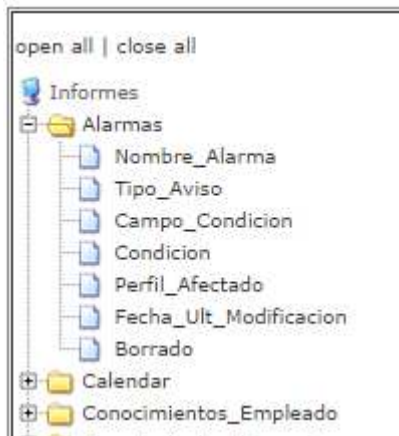


Figura 72 - Ejemplo visualización de datos con Dtree

- JQuery y Datatables - Utilizadas para mostrar los datos de la aplicación en formato de tablas, con opciones de filtrado y búsqueda

10 registros por página		Búsqueda		
Alarma	Tipo de Aviso Alarma	Campo Condicion	Dias	Empleados Afectados
Alarma1	Correo	Ini Macrodisenio	5	Jefe Proyecto
Alarma2	Mensaje en Sistem	Ini Oferta	-4	Jefe Proyecto
Alarma3	Aviso Responsable	Ini Macrodisenio	-1	Analista
Mostrando del 1 al 3 de 3 registros				
Primera Pag. Anterior .1 Siguiente Última Pag.				

Figura 73 - Ejemplo visualización de datos con jquery y datatables

- jscalendar - Utilizada para mostrar un calendario al trata de introducir una fecha en el sistema.

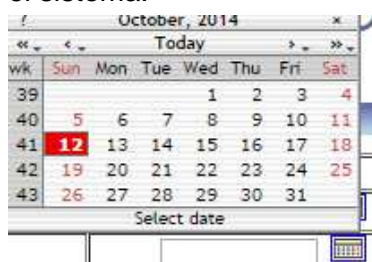


Figura 74 - Ejemplo visualización de datos con jscalendar

## 7 - Pruebas de Sistema

Todo programa informático, una vez finalizada su construcción ha de enfrentarse a una serie de pruebas que verifiquen que el comportamiento del mismo es el esperado, y no existe ningún tipo de error dentro del código desarrollado.

Estas pruebas se basan en un conjunto finito de casos de prueba, las cuales estarán seleccionadas con la intención por un lado de asegurar que los apartados más críticos del programa funcionan correctamente, y por otro lado, con la intención de sacar posibles fallos al programa

Los objetivos básicos de las pruebas son:

- Detectar defectos en el software.
- Verificar la integración adecuada de los componentes.
- Verificar que todos los requisitos se han implementado correctamente.
- Identificar y asegurar que los defectos encontrados se han corregido antes de entregar el software al cliente.
- Diseñar casos de prueba que sistemáticamente saquen a la luz diferentes clases de errores, haciéndolo con la menor cantidad de tiempo y esfuerzo.

El cuadro de pruebas de la aplicación estará formado por los siguientes elementos:

- Test Case --> Identificador único de la prueba
- Descripción --> Breve descripción de la prueba que ha de realizarse
- Precondiciones --> Condiciones que han de cumplirse para que la prueba pueda llevarse a cabo
- Resultado esperado --> Resultado que ha de darse para que la prueba pueda darse como Ok
- Resultado --> Resultado Final obtenido en la prueba



A continuación se presenta el cuadro de pruebas que ha sido ejecutado para la verificación del correcto funcionamiento de esta aplicación:

Test Case	Descripción	Precondiciones	Resultado de prueba esperado	Resultado de la prueba
TC-1	Logado en el sistema con usuario correcto	Usuario previamente dado de alta en el sistema	Usuario logado en el sistema. Mensaje de correcto logado en el sistema, y redirección a la página de inicio de la aplicación	OK
TC-2	Logado en el sistema con inserción errónea de password	Usuario previamente dado de alta en el sistema	Usuario no logado en el sistema. Mensaje de credenciales incorrectas y redirección de nuevo a la página de logado.	OK
TC-3	Logado en el sistema con usuario inexistente	Usuario no dado de alta en el sistema previamente	Mensaje de credenciales incorrectas y redirección de nuevo a la página de logado	OK
TC-4	Mensajes de Sistema - Acceso a la funcionalidad de mensajería en el sistema	Usuario correctamente logado y sin mensajes en el sistema	Correcto acceso a la página de mensajes, sin mensajes	OK
TC-5	Acceso a la funcionalidad de mensajería en el sistema con mensajes	Usuario correctamente logado y con mensajes en el sistema	Aparece la pagina de mensajería, y los mensajes pendientes de leer marcados como No Leídos	OK
TC-6	Mensajes de Sistema - Eliminación de un mensaje del sistema	Usuario correctamente logado y con mensajes en el sistema	Mensaje eliminado de la lista de mensajes de sistema	OK
TC-7	Mensajes de Sistema - Marcado de un mensaje del sistema como leído	Usuario correctamente logado y con mensajes en el sistema pendiente de leer	El mensaje pasa a estar marcado como leído	OK
TC-8	Proyecto - Alta de Nuevo proyecto en BBDD	Código de proyecto no existente previamente	El proyecto es creado correctamente en BBDD en la tabla Proyecto	OK
TC-9	Proyecto - Alta de proyecto ya existente en BBDD	Código de proyecto ya dado de alta en el sistema	Se redirige a pagina de error indicando que ya existe un código de proyecto con el mismo código	OK
TC-10	Proyecto - Planificación de un proyecto en BBDD	Código de proyecto ya dado de alta en el sistema	Se crea correctamente la planificación en BBDD en la tabla Planificacion_Proyecto.	OK

Test Case	Descripción	Precondiciones	Resultado de prueba esperado	Resultado de la prueba
TC-11	Proyecto - Inserción de un nuevo conocimiento a un proyecto	Conocimiento no existente para proyecto previamente dado de alta en el sistema	El conocimiento queda registrado en BBDD en la tabla Conocimientos_Proyecto	OK
TC-12	Proyecto - Autoplanificación de un proyecto	Proyecto previamente dado de alta en el sistema y planificado	Aparece la pantalla de planificación con los logines recomendados y en caso de aceptar, se almacena correctamente en la tabla Rel_Proyecto_Empleado cada asignación	OK
TC-13	Proyecto - Planificación manual de un proyecto	Proyecto previamente dado de alta en el sistema y planificado	Aparece la pantalla de planificación sin logines y con desplegables seleccionables y en caso de aceptar, se almacena correctamente en la tabla Rel_Proyecto_Empleado cada asignación	OK
TC-14	Proyecto - En modificación de proyecto, búsqueda de proyecto existente por código y nombre	Código de proyecto existente en BBDD.	Se redirige a la pantalla con la información del proyecto	OK
TC-15	Proyecto - Correcta modificación de proyecto	Proyecto previamente dado de alta	Proyecto modificado correctamente en BBDD en la tabla Proyecto.	OK
TC-16	Proyecto - Correcta subida de nueva documentación al servidor	Proyecto previamente dado de alta	Aparece la pantalla de archivo subido correctamente, y este se encuentra localizado dentro del servidor.	OK
TC-17	Proyecto - Correcta modificación de planificación	Proyecto previamente dado de alta en el sistema y planificado	Planificación correctamente modificada en Base de datos.	OK
TC-18	Proyecto - Correcta modificación de conocimientos de proyecto	Proyecto previamente dado de alta en el sistema, y con conocimientos	Conocimientos correctamente modificados en Base de datos.	OK
TC-19	Proyecto - Correcta reasignación de recursos	Proyecto previamente dado de alta en el sistema, y con recursos	Proyecto reasignado a los nuevos recursos y eliminados de los recursos anteriores.	OK
TC-20	Proyecto - Buscar proyecto - de proyecto, búsqueda de proyecto existente por código y nombre	Proyecto previamente dado de alta en el sistema	Se redirige a la pantalla con la información del proyecto	OK

Test Case	Descripción	Precondiciones	Resultado de prueba esperado	Resultado de la prueba
TC-21	Proyecto - Correcto Listado de documentación en servidor para el proyecto	Proyecto previamente dado de alta en el sistema y con documentación	Se listan correctamente los documentos del servidor para el proyecto.	OK
TC-22	Proyecto - Correcta descarga de archivos desde servidor para el proyecto	Proyecto previamente dado de alta en el sistema y con documentación	Se listan correctamente los documentos del servidor para el proyecto.	OK
TC-23	Proyecto - Borrado de documentación del servidor	Proyecto previamente dado de alta en el sistema y con documentación subida	Se borra correctamente el documento del servidor para el proyecto	OK
TC-24	Proyecto - Eliminación de proyecto inexistente por código y nombre	Proyecto no dado de alta previamente en el sistema	No se realiza ninguna acción y aparece mensaje de error	OK
TC-25	Proyecto - Eliminación de proyecto existente	Proyecto previamente dado de alta en el sistema	Se espera que el proyecto sea marcado como Borrado en base de datos	OK
TC-26	Proyecto - Cierre de proyecto existente	Proyecto previamente dado de alta en el sistema	Se espera que el proyecto sea marcado como cerrado y la puntuación del mismo se actualice.	OK
TC-27	Empleados - Visualización correcta de los datos propios	Empleado dado de alta previamente en el sistema	Se espera que aparezcan los datos del empleado logado por pantalla, y que pueda actualizar correctamente la password	OK
TC-28	Empleados - Visualización correcta de proyectos asignados	Empleado dado de alta previamente en el sistema y proyectos asignados	Se espera que aparezcan los proyectos asignados al usuario logado por pantalla.	OK
TC-29	Empleados - Reporte de horas correcto	Empleado dado de alta previamente en el sistema, reporte de horas sin insertar	Se espera que el usuario logado pueda insertar correctamente reportes de horas	OK
TC-30	Empleados - Reporte de horas duplicado	Empleado dado de alta previamente en el sistema y reporte de horas y a insertado	Se espera que el usuario logado no pueda insertar el reporte y aparezca un mensaje de error en pantalla.	OK

Test Case	Descripción	Precondiciones	Resultado de prueba esperado	Resultado de la prueba
TC-31	Empleados - Visualización correcta de reportes ya realizados.	Empleado dado de alta previamente en el sistema y reportes insertados	Se espera que aparezcan por pantalla todos los reportes del usuario logado	OK
TC-32	Empleados - Creación correcta de solicitud	Empleado dado de alta previamente en el sistema	Se espera que la solicitud se cree correctamente y aparezca en BBDD.	OK
TC-33	Empleados - Visualización correcta de solicitudes pendientes	Empleado dado de alta previamente en el sistema y solicitudes insertadas	Se espera que aparezcan por pantalla las solicitudes pendientes para el usuario	OK
TC-34	Empleados - Visualización correcta de solicitudes aprobadas/rechazadas	Empleado dado de alta previamente en el sistema y solicitudes aprobadas/rechazadas	Se espera que aparezcan por pantalla las solicitudes pendientes para el usuario.	OK
TC-35	Empleados - Aprobación/rechazo correcta de solicitudes pendientes	Empleado dado de alta previamente en el sistema y con solicitudes pendientes de aprobar.	Se espera que pueda aprobar, rechazar solicitudes	OK
TC-36	Alarmas - Creación correcta de nueva alarma	Alarma no existente previamente	Se espera que la alarma se pueda crear correctamente y aparezca reflejada en Base de datos	OK
TC-37	Alarmas - Error en la creación de alarma existente	Alarma existente previamente en BBDD	Se espera que la alarma no pueda ser insertada, aparezca un mensaje en pantalla y no se generen duplicados.	OK
TC-38	Alarmas - Modificación correcta de alarma	Alarma existente previamente en BBDD	Se espera que la alarma pueda ser modificada correctamente y aparezca cambiada en BBDD.	OK
TC-39	Alarmas - Eliminación correcta de alarma	Alarma existente previamente en BBDD	Se espera que la alarma pueda ser eliminada correctamente y aparezca como borrada en BBDD.	OK
TC-40	Alarmas - Listado correcto de todas la alarmas del sistema	Alarmas existentes previamente en BBDD	Se espera que aparezca el listado de las alarmas activas.	OK

Test Case	Descripción	Precondiciones	Resultado de prueba esperado	Resultado de la prueba
TC-41	Alarmas - Correcta generación de correo al cumplirse una alarma	Alarmas existentes previamente en BBDD y demonio en ejecución	Se espera que el usuario reciba un correo con los datos de la alarma	OK
TC-42	Alarmas - Correcta generación de mensaje en sistema al cumplirse una alarma	Alarmas existentes previamente en BBDD y demonio en ejecución	Se espera que el usuario reciba un mensaje en el sistema con los datos de la alarma	OK
TC-43	Informes - Creación de nuevo informe correcto seleccionando varios campos del árbol desplegable de campos.	N/A	Se espera que se genere un informe con los datos seleccionados correctamente	OK
TC-44	Informes - Guardado correcto de informe como Favorito	Informe previamente generado.	Se espera que el informe sea almacenado en BBDD con el nombre indicado	OK
TC-45	Informes - Error al guardar de informe como Favorito por existir previamente.	Informe previamente generado y guardado con el mismo nombre	Se espera que el informe no sea almacenado por duplicado en BBDD y aparezca un mensaje de error por pantalla	OK
TC-46	Informes - Exportación correcta de informe a excel	Informe previamente generado.	Se espera que el informe sea exportado correctamente en el repositorio de informes del usuario	OK
TC-47	Informes - correcta visualización de sql de informe en la generación	Informe previamente generado	Se espera que en la pantalla de sql, aparezca correctamente la sql autogenerada	OK
TC-48	Informes - Correcta visualización de informes favoritos	Informes favoritos almacenados	Se espera que aparezcan por pantalla todos los informes favoritos del usuario.	OK
TC-49	Informes - Correcta eliminación de informes favoritos	Informes favoritos almacenados.	Se espera que el informe seleccionado sea marcado como Borrado en BBDD.	OK
TC-50	Informes - Reejecución correcta de informe favorito	Informes favoritos almacenados.	Se espera que el informe seleccionado sea reejecutado correctamente.	OK
TC-51	Informes - correcta planificación de informe existente	Informes favoritos almacenados	Se espera que se planifique correctamente y aparezca dicha planificación en BBDD	OK
TC-52	Informes - Error de planificación de informe ya planificado	Informes favoritos almacenados y ya planificado.	Se espera que se aparezca mensaje de error de informe ya planificado y no se generen duplicados.	OK
TC-53	Informes - Correcto listado de informes planificados	Informes favoritos almacenados y ya planificados	Se espera que se aparezca el listado de todos los informes planificados	OK

Test Case	Descripción	Precondiciones	Resultado de prueba esperado	Resultado de la prueba
TC-54	Informes - Comprobación de correcta generación de informes mediante demonio de informes diarios, semanales, mensuales, y una vez	Informes favoritos almacenados y ya planificados, y demonio de ejecución corriendo.	Se espera que se aparezca el listado de todos los informes planificados.	OK
TC-55	Informes - Correcto listado de informes existentes en el repositorio para el usuario	Informes existentes en el repositorio	Se espera que se listen correctamente los informes para el usuario.	OK
TC-56	Administración - Correcta alta de nuevo empleado en BBDD	Empleado no existente en BBDD	Se espera que el empleado sea correctamente dado de alta en BBDD.	OK
TC-57	Administración - Error en el alta de un empleado ya existente	Empleado existente en BBDD	Se espera se devuelva error al insertar el empleado y no se generen duplicados en BBDD	OK
TC-58	Administración - Inserción de un nuevo conocimiento al empleado	Empleado existente en BBDD	Se espera se inserten correctamente los conocimientos para el empleado	OK
TC-59	Administración . En modificación, correcta búsqueda de empleado existente por nombre y apellidos	Empleado existente en BBDD	Se espera que aparezcan los datos del empleado	OK
TC-60	Administración .Correcta modificación de empleado	Empleado existente en BBDD	Se espera que los datos del empleado sean actualizados correctamente	OK
TC-61	Administración - Correcta modificación de conocimiento de empleado	Empleado y conocimiento existente en BBDD	Se espera que los datos del conocimiento del empleado sean actualizados correctamente	OK
TC-62	Administración - Correcta eliminación de empleado	Empleado existente en BBDD	Se espera el empleado sea correctamente marcado como Borrado de la BBDD.	OK
TC-63	Administración - Correcta actualización de derechos en BBDD	N/A	Se espera que los derechos sean actualizados correctamente en BBDD	OK
TC-64	Desconexión correcta del sistema	Usuario previamente logado	Se espera que el usuario sea correctamente deslogado del sistema y redirigido a la página de logado	OK
TC-65	Con usuario no autorizado a utilizar una funcionalidad, trata de conectar a acceder a ella	Usuario previamente logado con perfil no capaz de utilizar una funcionalidad, y otro con capacidad de utilizarla	Se espera que el usuario que no puede acceder no tenga acceso a dicha funcionalidad, y que el usuario que si puede acceder, tenga acceso	OK

Dentro del anexo " 11.1 - Anexo Pruebas Sistema Gestión Cartera", se encuentra el detalle de cada una de las pruebas aquí definidas.

## 8 - Despliegue

Como se indicaba dentro del apartado 3.4.2.2 de este mismo documento, para realizar el despliegue de la aplicación, se ha optado por contratar unos servicios de "Cloud Computing", dentro de los servidores web de Amazon.

Dentro de estos servidores, podremos instalar las aplicaciones que son necesarias para el despliegue de esta aplicación, las cuales serán:

- Sistema operativo Ubuntu, desde el que podremos manejar nuestra aplicación
- Apache Tomcat, con el cual realizaremos el despliegue de nuestra aplicación en productivo.
- MySQL, motor de base de datos, en el que crearemos las tablas y se almacenarán los datos de la aplicación.

Para contratar los servicios, dar de alta nuestro nuevo servidor y realizar el despliegue, seguiremos los siguientes pasos:

- Creación de servidor.
- Configuración del servidor
- Despliegue de la aplicación.
- Creación de Base de datos, tablas y datos
- Configuración de directorios



## 8.1 - Creación de nuevo servidor

Crearemos una nueva cuenta de usuario dentro de Amazon web Service.

Para ello, accedemos a la dirección <http://aws.amazon.com/es/> , y nos crearemos una nueva cuenta de usuario.

Una vez creada, pincharemos en la pestaña Services y a su vez al subapartado EC2:

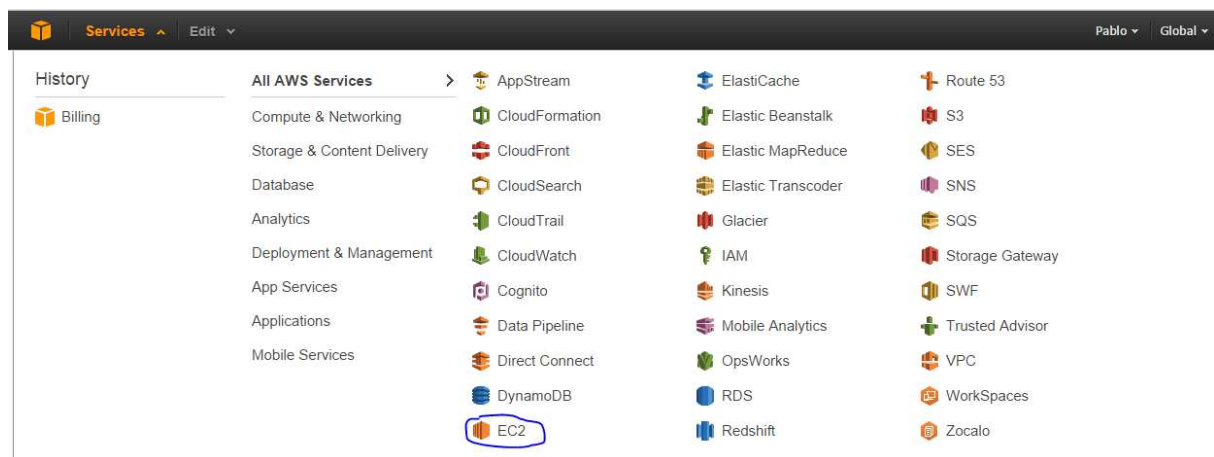


Figura 75- Pagina de selección de servicio en AWS

Seleccionaremos la región en la que queremos que esté el servidor. En este caso, y por cercanía, lo situaremos en la región de Irlanda:



Figura 76 - Selección de localización de servidor en AWS

Una vez seleccionada la región , crearemos una nueva instancia que será nuestro servidor, en el cual tendremos nuestra aplicación:

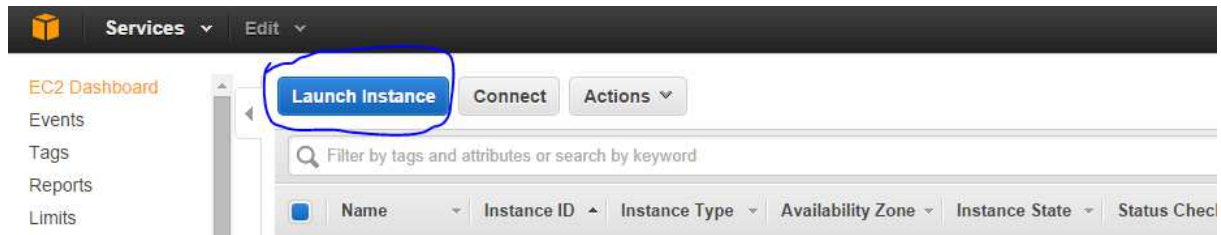


Figura 77 - Creación de nueva instancia de servicio en AWS

Seleccionaremos el tipo de sistema operativo que esté disponible en nuestro servidor. En este caso y tal y como se detallo en la oferta al cliente, se ha seleccionado un sistema operativo Ubuntu 14.04

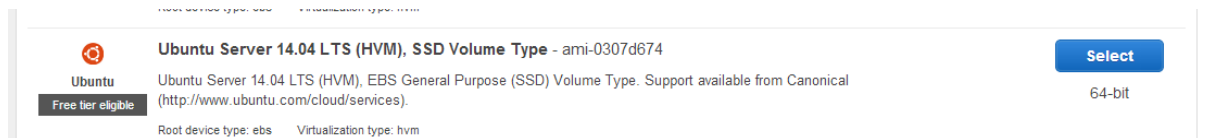


Figura 78 - Selección de sistema operativo en AWS

Seleccionaremos como queremos que sea nuestro servidor.

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
<input checked="" type="checkbox"/>	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate
<input type="checkbox"/>	General purpose	m3.medium	1	3.75	1 x 4 (SSD)	-	Moderate
<input type="checkbox"/>	General purpose	m3.large	2	7.5	1 x 32 (SSD)	-	Moderate

Figura 79 - Selección de tipo de servidor en AWS

Una vez seleccionado, ya tenemos nuestro servidor creado, apareciéndonos en pantalla la correcta creación del mismo:

### Launch Status

✓ **Your instance is now launching**

The following instance launch has been initiated: [i-e770e6a7](#) [View launch log](#)

💬 **Get notified of estimated charges**

[Create billing alerts](#) to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

**How to connect to your instance**

Your instance is launching, and it may take a few minutes until it is in the **running** state, when it will be ready for you to use. Usage hours on your new instance will start immediately and continue to accrue until you stop or terminate your instance.

Click **View Instances** to monitor your instance's status. Once your instance is in the **running** state, you can **connect** to it from the instances screen. [Find out](#) how to connect to your instance.

▼ **Here are some helpful resources to get you started**

- [How to connect to your Linux instance](#)
- [Learn about AWS Free Usage Tier](#)
- [Amazon EC2: User Guide](#)
- [Amazon EC2: Discussion Forum](#)

While your instances are launching you can also

Figura 80 - Pagina de correcta creación de servidor en AWS

En este momento, ya tendremos nuestra instancia creada dentro de Amazon:

Instances

Spot Requests

Reserved Instances

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK STORE

Launch Instance

Connect

Actions ▼

Filter by tags and attributes or search by keyword

1 to 1 of 1

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS	Public IP
	i-e770e6a7	t2.micro	eu-west-1a	running	2/2 checks ...	None	ec2-54-76-232-76.eu-w...	54.76.232.71

Figura 81 - Instancia correctamente creada en AWS

## 8.2 - Configuración del servidor

Para poder utilizar el servidor vía HTTP , FTP, etc, es necesario que abramos los puertos necesarios, para que este pueda ser accedida, mediante la modificación de los parámetros del grupo de seguridad.

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ
Custom TCP Rule	TCP	8080	0.0.0.0/0
SSH	TCP	22	0.0.0.0/0
HTTP	TCP	80	0.0.0.0/0
Custom TCP Rule	TCP	21	0.0.0.0/0
Custom TCP Rule	TCP	1024 - 1048	0.0.0.0/0

Figura 82 - Apertura de puertos en AWS

Una vez abiertos los puertos, ya tenemos acceso mediante ssh al servidor, por lo que podremos instalarle los programas necesarios.

Para ello, accedemos mediante Putty vía ssh, con la clave publica que ha sido generada en la instalación del servidor, y mediante su conversión a una clave entendible por putty con el putty keygen.

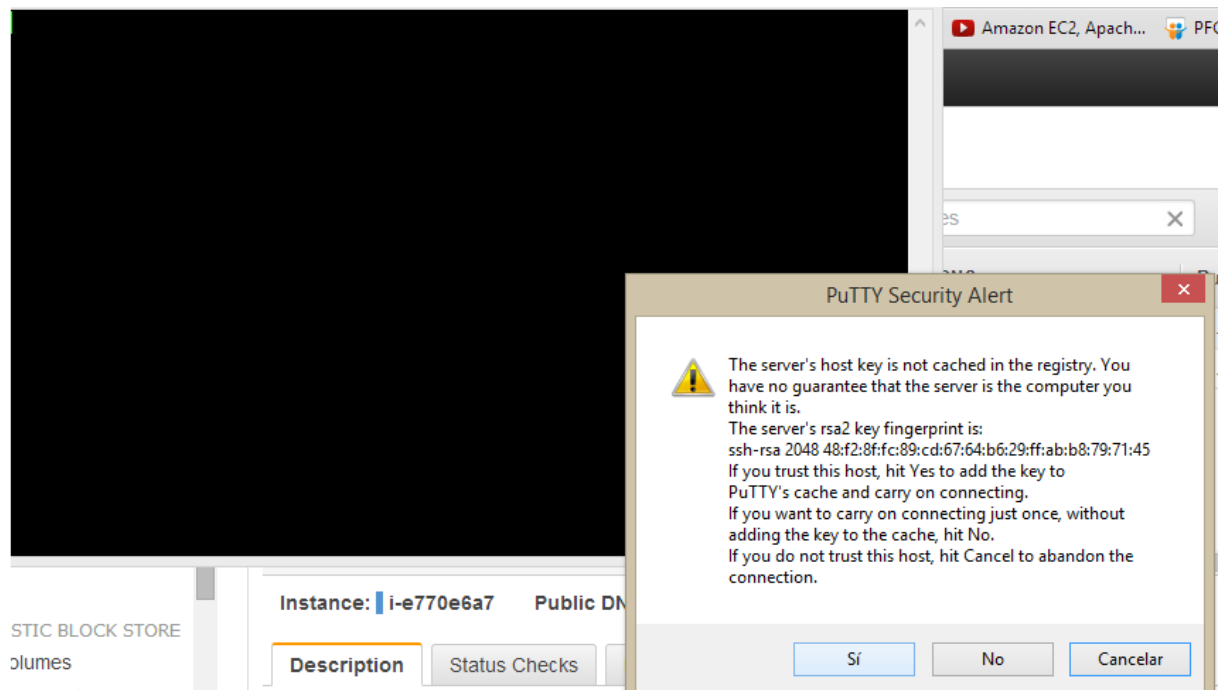


Figura 83 - Conexión mediante putty al nuevo servidor mediante sftp

Una vez dentro del servidor, escribimos los siguientes comandos, para realizar la actualización del sistema operativo (sudo aptitude update , sudo aptitude upgrade):

```
ubuntu@ip-172-31-25-254:~$ sudo aptitude update
Ign http://eu-west-1.ec2.archive.ubuntu.com trusty InRelease
Ign http://eu-west-1.ec2.archive.ubuntu.com trusty-updates InRelease
Get: 1 http://eu-west-1.ec2.archive.ubuntu.com trusty Release.gpg [933 B]
Get: 2 http://eu-west-1.ec2.archive.ubuntu.com trusty-updates Release.gpg [933 B]
Get: 3 http://eu-west-1.ec2.archive.ubuntu.com trusty Release [58.5 kB]
Get: 4 http://eu-west-1.ec2.archive.ubuntu.com trusty-updates Release [59.7 kB]
Ign http://security.ubuntu.com trusty-security InRelease
Get: 5 http://eu-west-1.ec2.archive.ubuntu.com trusty/main Sources [1,064 kB]
Get: 6 http://security.ubuntu.com trusty-security Release.gpg [933 B]
Get: 7 http://eu-west-1.ec2.archive.ubuntu.com trusty/universe Sources [6,399 kB]
Get: 8 http://security.ubuntu.com trusty-security Release [59.7 kB]
Get: 9 http://eu-west-1.ec2.archive.ubuntu.com trusty/main amd64 Packages [1,350 kB]
Get: 10 http://security.ubuntu.com trusty-security/main Sources [36.3 kB]
Get: 11 http://security.ubuntu.com trusty-security/universe Sources [8,431 B]
Get: 12 http://eu-west-1.ec2.archive.ubuntu.com trusty/universe amd64 Packages [5,859 kB]
Get: 13 http://security.ubuntu.com trusty-security/main amd64 Packages [119 kB]
Get: 14 http://eu-west-1.ec2.archive.ubuntu.com trusty-updates/main Sources [102 kB]
Get: 15 http://eu-west-1.ec2.archive.ubuntu.com trusty-updates/universe Sources [62.6 kB]
Get: 16 http://security.ubuntu.com trusty-security/universe amd64 Packages [39.2 kB]
Get: 17 http://eu-west-1.ec2.archive.ubuntu.com trusty-updates/main amd64 Packages [277 kB]
Get: 18 http://eu-west-1.ec2.archive.ubuntu.com trusty-updates/universe amd64 Packages [163 kB]
Get: 19 http://security.ubuntu.com trusty-security/main Translation-en [56.9 kB]
Get: 20 http://eu-west-1.ec2.archive.ubuntu.com trusty-updates/main Translation-en [119 kB]
Get: 21 http://eu-west-1.ec2.archive.ubuntu.com trusty-updates/universe Translation-en [79.9 kB]
Get: 22 http://security.ubuntu.com trusty-security/universe Translation-en [22.2 kB]
Ign http://eu-west-1.ec2.archive.ubuntu.com trusty/main Translation-en_US
Ign http://eu-west-1.ec2.archive.ubuntu.com trusty/main Translation-en
Ign http://eu-west-1.ec2.archive.ubuntu.com trusty/universe Translation-en_US
Ign http://eu-west-1.ec2.archive.ubuntu.com trusty/universe Translation-en
Fetched 15.9 MB in 4s (3,273 kB/s)

Current status: 54 updates [+54], 127 new [+127].
```

Figura 84 - Actualización de funciones principales en el servidor

```
ubuntu@ip-172-31-25-254:~$ sudo aptitude upgrade
Resolving dependencies...
The following NEW packages will be installed:
  linux-headers-3.13.0-32(a) linux-headers-3.13.0-32-generic(a)
The following packages will be upgraded:
  accountsservice apt apt-transport-https apt-utils base-files bsdutils cloud-init dbus dpkg file gnupg gpgv grub-common grub-legacy-ec2 grub-pc grub-pc-bin
  grub2-common initramfs-tools initramfs-tools-bin language-selector-common libaccountsservice0 libapt-inst1.5 libapt-pkg4.12 libblkid1 libboost-iostreams1.54.0
  libboglmanager0 libdbus-1-3 libjson-c2 libjson0 libmagic1 libmount1 libpam-systemd libsel1.0.0 libsystemd-daemon0 libsystemd-login0 libtasn1-6 libudev1 libuuid1
  libxml2 linux-headers-generic linux-headers-virtual linux-image-virtual linux-virtual mount open-vm-tools openssl python3-gi resolvconf systemd-services tzdata
  udev upstart util-linux uuid-runtime
54 packages upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 38.8 MB of archives. After unpacking 119 MB will be used.
Do you want to continue? [Y/n/?] y
Get: 1 http://eu-west-1.ec2.archive.ubuntu.com/ubuntu/ trusty-updates/main base-files amd64 7.2ubuntu5.1 [70.2 kB]
Get: 2 http://eu-west-1.ec2.archive.ubuntu.com/ubuntu/ trusty-updates/main dpkg amd64 1.17.5ubuntu5.3 [1,954 kB]
Get: 3 http://eu-west-1.ec2.archive.ubuntu.com/ubuntu/ trusty-updates/main mount amd64 2.20.1-5.1ubuntu20.1 [115 kB]
Get: 4 http://eu-west-1.ec2.archive.ubuntu.com/ubuntu/ trusty-updates/main tzdata all 2014e-0ubuntu0.14.04 [131 kB]
Get: 5 http://eu-west-1.ec2.archive.ubuntu.com/ubuntu/ trusty-updates/main util-linux amd64 2.20.1-5.1ubuntu20.1 [456 kB]
Get: 6 http://eu-west-1.ec2.archive.ubuntu.com/ubuntu/ trusty-updates/main libapt-pkg4.12 amd64 1.0.1ubuntu2.1 [639 kB]
Get: 7 http://eu-west-1.ec2.archive.ubuntu.com/ubuntu/ trusty-updates/main gpgv amd64 1.4.16-1ubuntu2.1 [160 kB]
```

Figura 85 - Actualización de funciones principales en el servidor

Con este tenemos el sistema operativo actualizado, con lo que podemos proceder a instalar los programas que son requeridos para nuestro programa. Esto lo hacemos mediante la llamada al tasksel:

```
ubuntu@ip-172-31-25-254:~$ sudo tasksel
```

Figura 86 - Comando de selección de paquetes a instalar

Lo que nos abre una nueva pantalla, en la que seleccionaremos :

- Basic Ubuntu Server
- Open SSH Server
- LAMP\_server --> con lo que instalaremos mysql entre otras
- Tomcat Java Server --> herramienta con la que realizaremos el despliegue

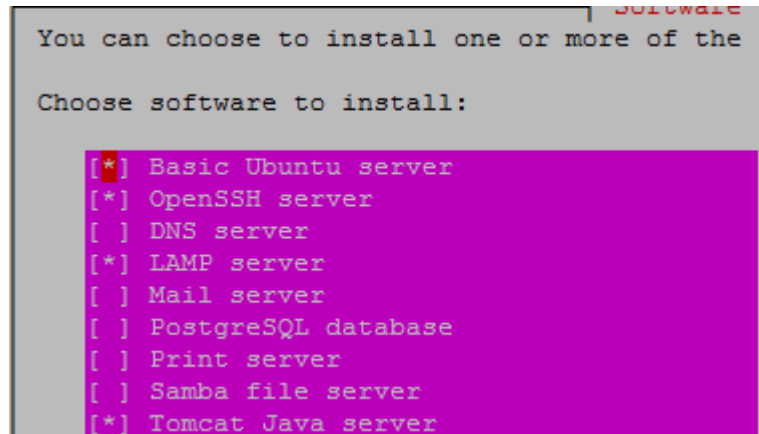


Figura 87 - Selección de aplicaciones a instalar dentro del servidor

Durante la instalación, se nos solicitará la password del usuario root para mysql :

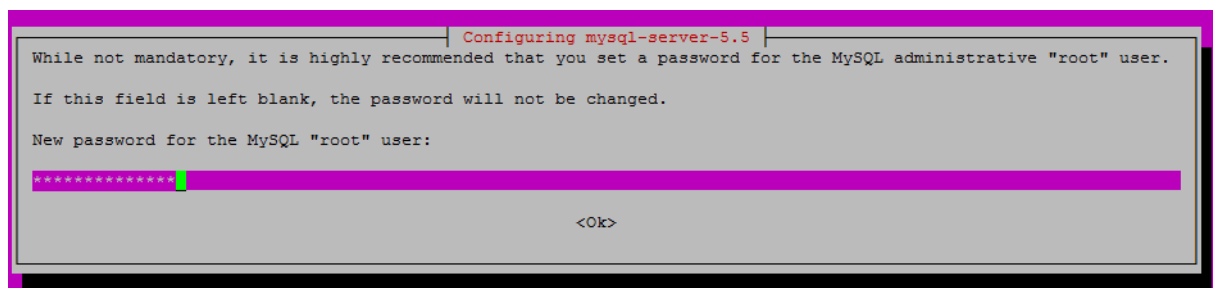


Figura 88 - Inserción de password para usuario root de Mysql

De cara a no tener que realizar el acceso mediante la inclusión del puerto en la dirección ip del servidor, crearemos una tabla de pivotaje que nos redirigirá siempre al puerto 8080 del servidor:

```
ubuntu@ip-172-31-25-254:~$ sudo iptables -t nat -I PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 8080
ubuntu@ip-172-31-25-254:~$ sudo iptables-save
# Generated by iptables-save v1.4.21 on Sun Aug 3 18:38:04 2014
*nat
:PREROUTING ACCEPT [0:0]
:INPUT ACCEPT [0:0]
:OUTPUT ACCEPT [2:241]
:POSTROUTING ACCEPT [2:241]
-A PREROUTING -p tcp -m tcp --dport 80 -j REDIRECT --to-ports 8080
COMMIT
# Completed on Sun Aug 3 18:38:04 2014
```

Figura 89 - Tabla de pivotaje para redireccionamiento de puertos

Finalmente para poder realizar el despliegue, nos crearemos un usuario de acceso al apache tomcat, herramienta con la que cargaremos el .war del proyecto. :

```
ubuntu@ip-172-31-25-254:~$ sudo vim /etc/tomcat7/tomcat-users.xml
distributed under the license is distributed on an "AS IS" basis,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<tomcat-users>
<user username="root" password=" " roles="manager-gui,admin-gui"/>
NOTE: By default, no user is included in the "manager-gui" role required
to operate the "/manager/html" web application. If you wish to use this app,
you must define such a user - the username and password are arbitrary.
```

Figura 90 - Inclusión de usuario y password de tomcat

Y por si fuera necesario, actualizamos el administrador de apache tomcat:

```
ubuntu@ip-172-31-25-254:~$ sudo apt-get install tomcat7-admin
Reading package lists... Done
Building dependency tree
Reading state information... Done
tomcat7-admin is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

Figura 91 - Actualización de administrador tomcat

Finalmente reiniciamos el sistema tomcat, para que todos los cambios queden aplicados:

```
ubuntu@ip-172-31-25-254:~$ sudo /etc/init.d/tomcat7 restart
* Stopping Tomcat servlet engine tomcat7
* Starting Tomcat servlet engine tomcat7
```

Figura 92 - Reinicio de aplicación tomcat

Una vez hechos todos estos pasos, ya tenemos completamente configurado nuestro sistema, para poder realizar el despliegue de la aplicación.

## 8.3 - Despliegue de la Aplicación

En primer lugar accedemos a Tomcat, escribiendo en la barra de direcciones, la ip que ha sido asignada para nuestra aplicación y debería aparecernos una página indicando que el tomcat funciona y con un enlace al "manager webapp". En este caso 54.76.232.76.

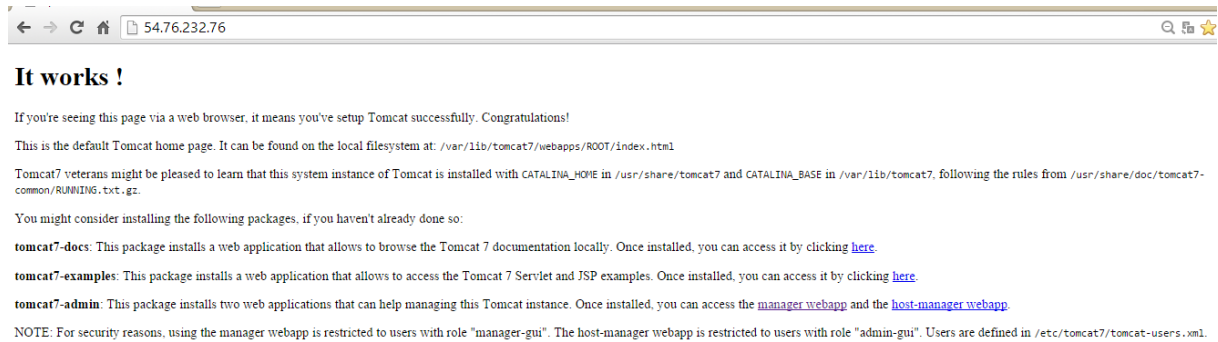


Figura 93 - Pagina de correcta instalación de tomcat

Al acceder al "manager webapp", nos solicitará la contraseña de administrador , que será la que hemos incluido en el archivo "tomcat-users.xml", dentro de la instalación:

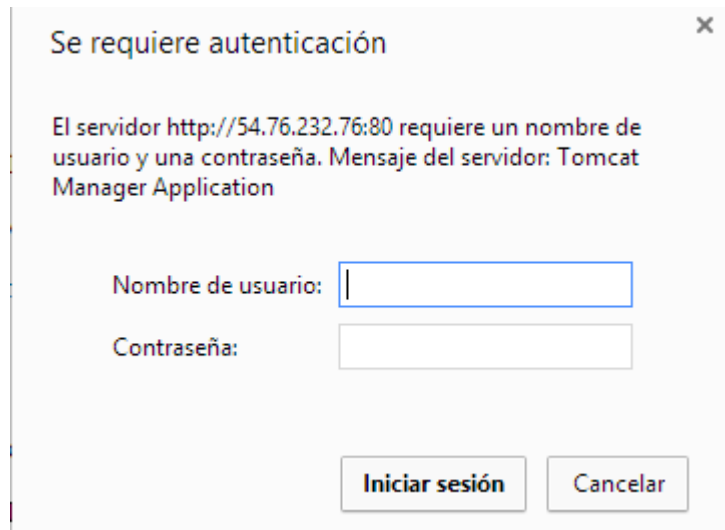


Figura 94 - Autenticación en Tomcat



Una vez metida la password, estaremos dentro del manager en donde seleccionaremos al archivo war a desplegar de nuestro proyecto:

Archivo WAR a desplegar

Seleccione archivo WAR a cargar

Seleccionar archivo

GestionCarteraPRR.war

Desplegar

Diagnósticos

Subiendo (11%)...

Figura 95 - Despliegue de proyecto war en Tomcat

Cuando ya haya sido desplegado, nuestro proyecto aparecerá dentro de la lista de aplicaciones desplegadas:

Mensaje:

OK

Gestor

Listar Aplicaciones

Ayuda HTML de Gestor

Ayuda de Gestor

Aplicaciones

Trayectoria	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguno especificado		true	0	<div>Arrancar Parar Recargar Replegar</div> <div>Expirar sesiones sin trabajar ≈ 30 minutos</div>
/GestionCarteraPRR	Ninguno especificado		true	0	<div>Arrancar Parar Recargar Replegar</div> <div>Expirar sesiones sin trabajar ≈ 30 minutos</div>
/docs	Ninguno especificado	Tomcat Documentation	true	0	<div>Arrancar Parar Recargar Replegar</div> <div>Expirar sesiones sin trabajar ≈ 30 minutos</div>
/examples	Ninguno especificado	Servlet and JSP Examples	true	0	<div>Arrancar Parar Recargar Replegar</div> <div>Expirar sesiones sin trabajar ≈ 30 minutos</div>
/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	0	<div>Arrancar Parar Recargar Replegar</div> <div>Expirar sesiones sin trabajar ≈ 30 minutos</div>
/manager	Ninguno especificado	Tomcat Manager Application	true	1	<div>Arrancar Parar Recargar Replegar</div> <div>Expirar sesiones sin trabajar ≈ 30 minutos</div>

Figura 96 - Correcto despliegue de la aplicación en el servidor

## 8.4 - Creación de base de datos, tablas y datos

Para poder acceder a Mysql, escribiremos el comando `mysql -u root -p`. Con esto nos solicitará la password, que hemos escrito a la instalación inicial del programa ejecutando el `taskel`:

```
ubuntu@ip-172-31-25-254:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 43
Server version: 5.5.38-0ubuntu0.14.04.1 (Ubuntu)

Copyright (c) 2000, 2014, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Figura 97 - Conexión a Mysql

Una vez dentro crearemos la base de datos que utilizaremos en el programa llamada "gestion":

```
mysql> create database gestion;
Query OK, 1 row affected (0.00 sec)
```

Figura 98 - Creación de Base de Datos con Mysql

Una vez creada, nos situaremos en nuestra nueva bbdd:

```
mysql> use gestion;
Database changed
```

Figura 99 - Cambio de Base de Datos en Mysql

Iremos creando las tablas, mediante un volcado de las tablas de desarrollo a producción. Para ello seguiremos los siguientes pasos:

**A - Exportamos los datos de desarrollo mediante la ejecución del siguiente comando por cada tabla de desarrollo:**

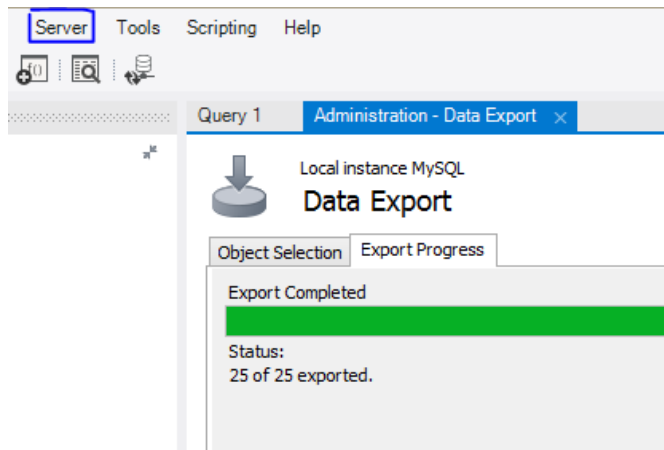


Figura 100 - Exportación de datos desde Mysql en desarrollo

De esta manera se generará un archivo sql por cada una de las tablas:

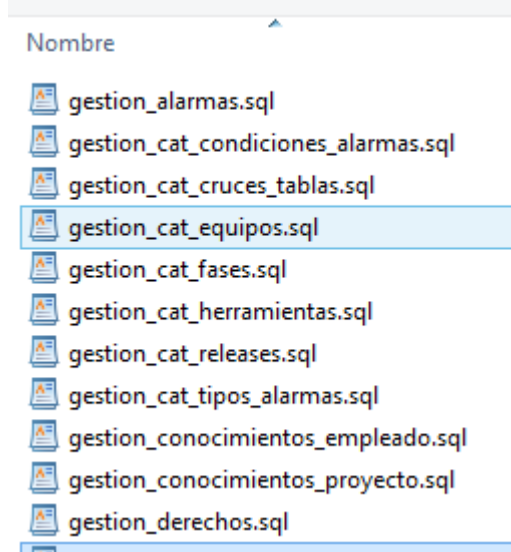


Figura 101 - Sqls generadas en la exportación de Mysql

**B - Subimos los archivos al servidor, mediante Filezilla :**

Servidor/Archivo local	Direcci...	Archivo remoto	Tamaño	Prioridad	Hora
↓ sftp://ubuntu@54.76.232.76					
<input type="checkbox"/> C:\Users\casa\Documents...	-->>	/home/ubuntu/sentencias sql...	2.024	Normal	28/09/2014 23:11:03
<input type="checkbox"/> C:\Users\casa\Documents...	-->>	/home/ubuntu/sentencias sql...	1.806	Normal	28/09/2014 23:11:03
<input type="checkbox"/> C:\Users\casa\Documents...	-->>	/home/ubuntu/sentencias sql...	1.822	Normal	28/09/2014 23:11:03
<input type="checkbox"/> C:\Users\casa\Documents...	-->>	/home/ubuntu/sentencias sql...	1.837	Normal	28/09/2014 23:11:03
<input type="checkbox"/> C:\Users\casa\Documents...	-->>	/home/ubuntu/sentencias sql...	1.793	Normal	28/09/2014 23:11:04
Archivos en cola    Transferencias fallidas <b>Transferencias satisfactorias (24)</b>					

Figura 102 - Subida de ficheros mediante Filezilla

**C- Pasamos todos los archivos a uno único concatenado :**

```
-rw-rw-r-- 1 ubuntu ubuntu 2015 Sep 28 23:12 gestion_reporte_horas.s
-rw-rw-r-- 1 ubuntu ubuntu 2027 Sep 28 23:12 gestion_solicitudes.sql
ubuntu@ip-172-31-18-9:~/sentencias sql$ cat * > salida_sql.sql
ubuntu@ip-172-31-18-9:~/sentencias sql$
```

Figura 103 - Concatenación de archivos de sql

**D -Cargamos el archivo salida\_sql.sql**

```
ubuntu@ip-172-31-18-9:~/sentencias sql$
ubuntu@ip-172-31-18-9:~/sentencias sql$ mysql -u root -p gestion < /home/ubuntu/sentencias_sql/salida_sql.sql
```

Figura 104 - Carga de archivo en el entorno de producción con mysql

Con esto, ya tendríamos todas las tablas creadas dentro de la BBDD tal y como se encontraban en el entorno de desarrollo.

Finalmente, insertamos el usuario de recursos humanos, dado que serán ellos los que carguen los datos de los distintos empleados que podrán utilizar la aplicación y sus perfiles, así como los datos de las tablas de funcionamiento de la aplicación. Esto se realizará mediante las inserciones incluidas dentro del punto "12.4 Anexo preparación base de datos"

## 8.5 - Configuración de directorios

De cara a poder realizar almacenar la documentación e informes, crearemos ambos directorios en un lugar en donde no se machaquen con cada despliegue de la aplicación. En este caso, en un directorio anterior al raíz de la aplicación:

```
ubuntu@ip-172-31-25-254:/var/lib/tomcat7$ sudo mkdir documentacion
ubuntu@ip-172-31-25-254:/var/lib/tomcat7$ sudo mkdir informes
```

Figura 105 - Creación de directorios de informes y documentación en servidor

Y finalmente le daremos permisos 777 para que todos los usuarios puedan acceder a dichos directorios:

```
ubuntu@ip-172-31-25-254:/var/lib/tomcat7$ sudo chmod 777 documentacion/
ubuntu@ip-172-31-25-254:/var/lib/tomcat7$ sudo chmod 777 informes/
```

Figura 106 - Cambio de permisos de directorios de informes y documentación en servidor

## 9 - Navegabilidad y Guía de Usuario

La navegabilidad de un sitio web podría definirse como el conjunto de propiedades de la interactividad del sitio web que permiten que el usuario del mismo sea capaz de moverse por su estructura e identificar las diferentes secciones y contenidos de una forma sencilla y efectiva y sin perderse en él. Es decir, podría medirse como la facilidad con la que un usuario puede desplazarse por todas las páginas que componen la aplicación.

Para poder alcanzar dicho objetivo, son varios los requisitos que han sido tenidos en cuenta a la hora de realizar esta aplicación:

- **Arquitectura Sencilla** - La web de una empresa no ha de contener grandes innovaciones tecnológicas como sliders, pop-up's, carga de contenido multimedia, etc., sino que ha de mantener una navegación simple y ha de estar diseñada para cumplir la función con la que se ha creado de la forma más rápida, sencilla, y fiable posible.
- **Diseño Amigable** - El diseño de una página web de empresa ha de evitar grandes inter cambios de colores, diseños entre páginas, etc, dado que está pensada para ser utilizada un gran número de horas al día.
- **Facilidad de Navegación** - La web ha de ser clara, con menús directos a las partes más importantes de la web, con la que luego podremos ir moviéndonos por los distintos subniveles desagregando la información de manera que el usuario sepa exactamente en todo momento como alcanzar el objetivo final propuesto.
- **Usabilidad y Claridad de Mensaje** - los mensajes devueltos por el sistema han de ser claros y la información debe estar distribuida para que se acceda a ella de una forma sencilla, intuitiva y mecánica.
- **Estructuración del sitio web** - La organización del sitio web, y como se reparten los contenidos dentro del mismo mediante agrupaciones, secciones, subsecciones, bien relacionados entre sí, hace que un sitio web sea legible y fácil de navegar por él.

## 9.1- Navegabilidad

El usuario de la aplicación podrá ir navegando desde la página principal de la aplicación, hasta cualquier punto de la web, mediante la selección de las distintas secciones que esta contiene.

Las funcionalidad que tendrá la aplicación, se basa en 5 secciones principales, a partir de las cuales cuelgan todas las opciones del sistema. Estas son :

- Cartera
- Empleados
- Alarmas
- Informes
- Administración

Estas secciones principales estarán siempre presentes dentro de toda la aplicación de cara a poder seleccionarlas desde cualquier punto de la misma:

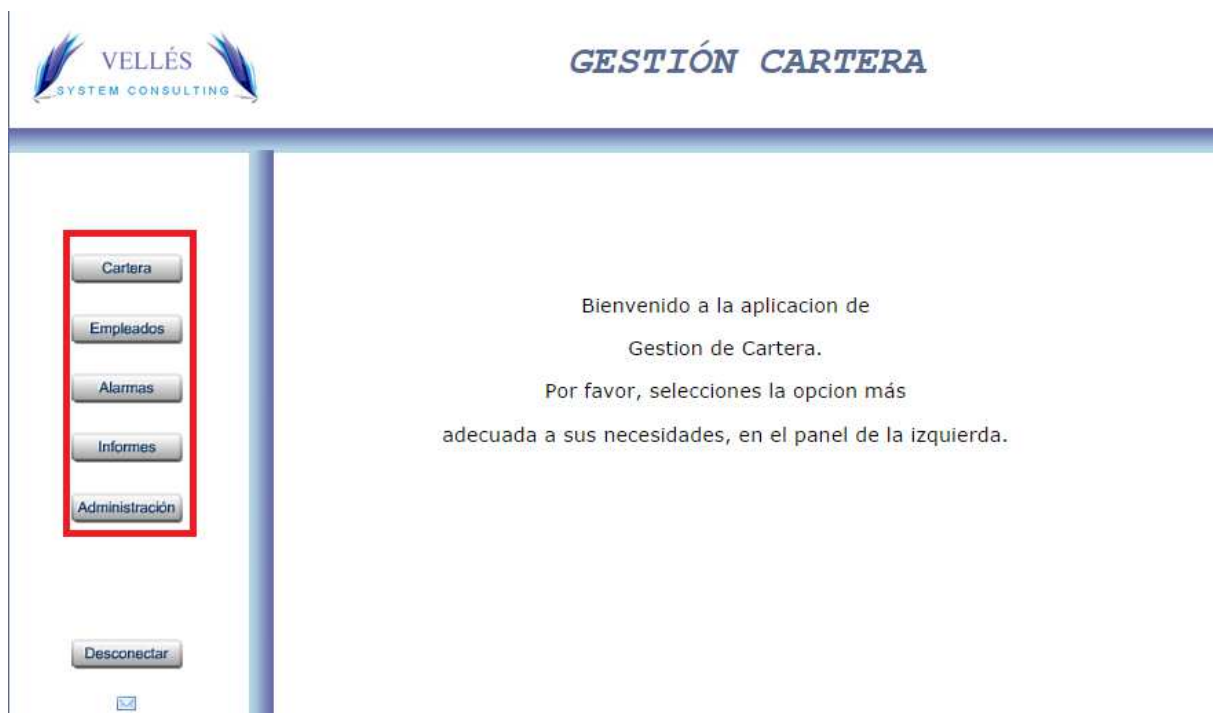


Figura 107 - Menú de navegación de aplicación

Las subsecciones que podremos ir seleccionando a partir de estas principales serán las siguientes:

## **Cartera**

Desde la sección de cartera podremos gestionar toda la cartera de proyectos de la empresa, desde su asignación, gestión, cambios de fase, asignación, etc.

La navegabilidad que estará disponible dentro del aplicación para esta sección se muestra mediante el siguiente diagrama:

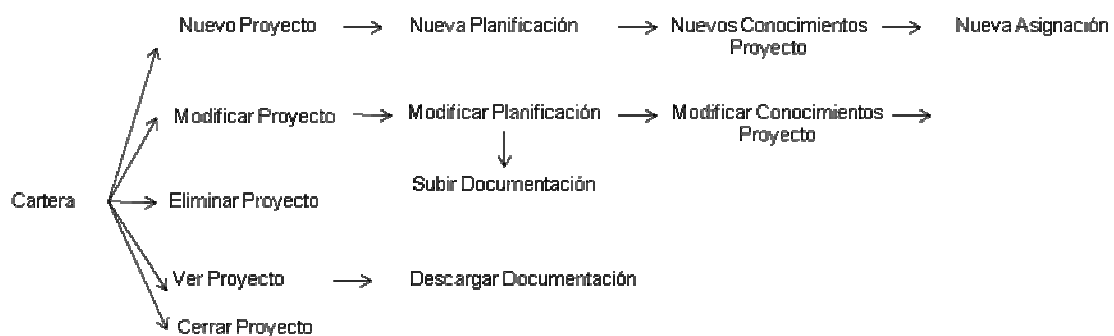


Figura 108 - Navegación de Menú Cartera

## **Empleados**

Desde la sección de empleados, los distintos empleados de la empresa podrán ver los datos que la empresa tiene sobre ellos, ver proyectos asignados, reportar horas, realizar solicitudes, etc.

La navegabilidad que estará disponible dentro del aplicación para esta sección se muestra mediante el siguiente diagrama:

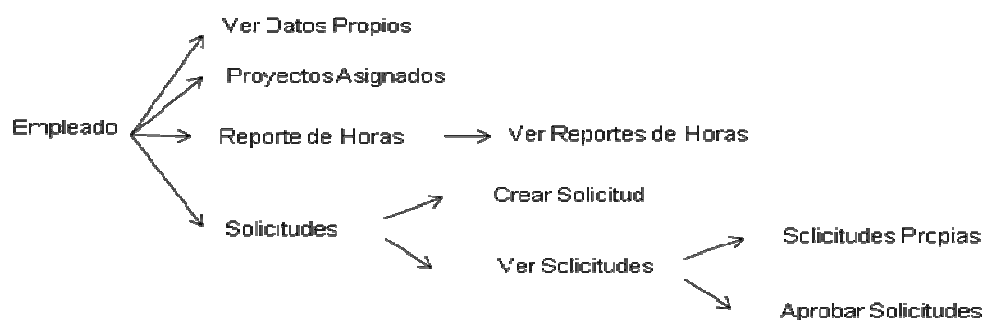


Figura 109- Navegación de Menú Empleado



## Alarmas

Desde la sección de alarmas, podremos crear alarmas que nos avisen del cumplimiento de las distintas fases de los proyectos, modificarlas, eliminarlas, etc.

La navegabilidad que estará disponible dentro de la aplicación para esta sección se muestra mediante el siguiente diagrama:

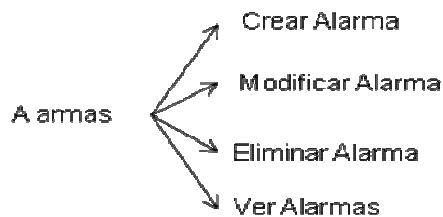


Figura 110 - Navegación de Menú Alarmas

## Informes

Desde la sección de informes, podremos realizar informes a partir de la información contenida dentro de las tablas de la aplicación, gestionar favoritos, planificar los informes para que sean ejecutados de manera nocturna evitando posibles esperas de obtención de información, acceder al repositorio de informes del propio usuario, etc.

La navegabilidad que estará disponible dentro de la aplicación para esta sección se muestra mediante el siguiente diagrama:

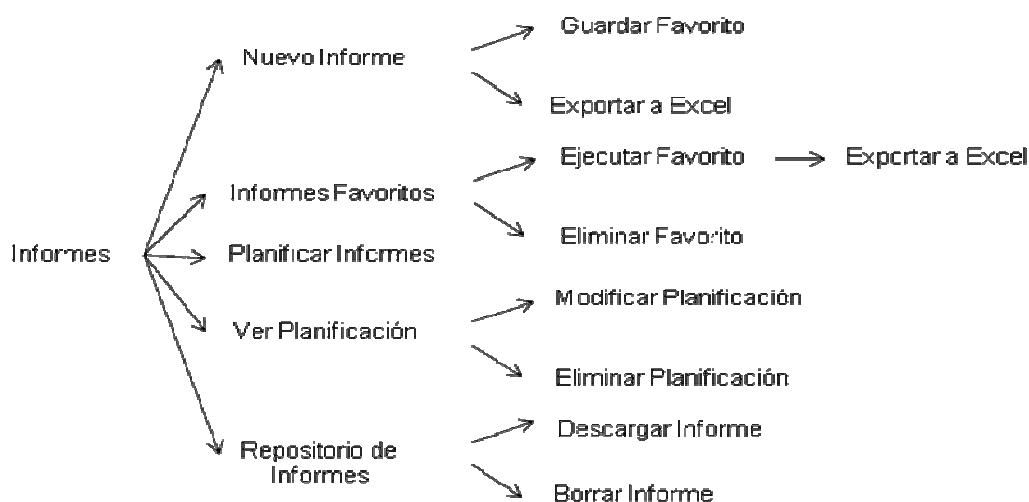


Figura 111 - Navegación de Menú Informes

## Administración

Desde la sección de administración, se podrá gestionar la cartera de empleados de la empresa, dando de alta, baja empleados, así como modificar los permisos de los distintos perfiles de acceso a las funcionalidades de la aplicación.

La navegabilidad que estará disponible dentro del aplicación para esta sección se muestra mediante el siguiente diagrama:

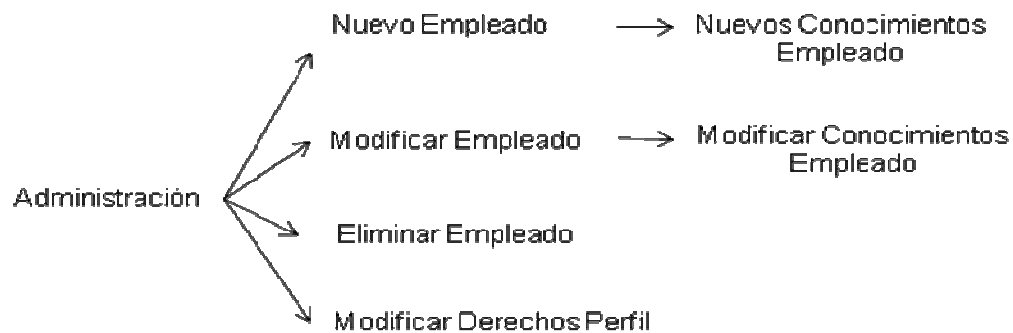


Figura 112 - Navegación de Menú Administración

## Mensajería en Sistema

Finalmente desde la página principal, el usuario podrá acceder a su buzón interno de mensajería, en el que recibirá distintas alarmas del sistema:



Figura 113 - Navegación de Mensajería Sistema

## 9.2 - Guía de Usuario

De cara a la explotación de la aplicación desarrollada y de cara a que los empleados de la empresa puedan aprender a manejarla, y utilizar todos los recursos que en ella se engloban, se hace necesaria la existencia de un manual o guía de usuario que muestre de una forma sencilla y clara dichas funciones, mediante la utilización tanto de texto escrito como de imágenes del sistema.

El manual de usuario de la aplicación Gestión Cartera, se encuentra integrado dentro del anexo "11.2 - Anexo Manual de Usuario".

## 10 - Presupuesto

La identificación y estimación de los costes y gastos empresariales que una empresa debe imputar a un proyecto que realiza es una tarea que va a determinar el precio de venta del producto o servicio o que nos va a permitir aceptar o rechazar la oferta de proyecto que nos proporciona el posible cliente. La realización de un proyecto lleva consigo una serie de costes directos e indirectos. Entendemos como costes directos aquellos que son imputables directamente al proyecto y consecuencia del mismo, como son los costes materiales, los costes de personal, las posibles contingencias. etc.

A lo largo de los siguientes capítulos, se detalla el coste que de cada uno de los conceptos que se generará para la realización de este proyecto, lo que nos llevará a obtener el precio mínimo por el que el proyecto podrá ser realizado.

### 10.1 - Recursos Humanos

Tal y como se detalló dentro del punto 4.2 de este mismo documento, para la realización de este proyecto será necesario contar con varios recursos a lo largo de su ciclo de vida de modo que este pueda ser realizado con éxito.

El coste que actualmente se marca dentro de la empresa para cada uno de los perfiles solicitados, es el siguiente:

Resource Name	Type	Standard Rate
Analista Funcional	Work	19,00 €/hr
Diseñador	Work	15,00 €/hr
Responsable de Calidad	Work	14,00 €/hr
Jefe de Proyecto	Work	24,00 €/hr
Programador1	Work	10,00 €/hr
Programador2	Work	10,00 €/hr

Figura 114 - Coste Hora de Recurso

De este modo, y con la planificación que se estableció dentro del punto 4.2 de este documento, tendremos un coste total por recurso, como sigue:

Resource Name	Cost
+ Analista Funcional	9.120,00 €
+ Diseñador	6.490,91 €
+ Responsable de Calidad	373,33 €
+ Jefe de Proyecto	19.680,00 €
+ Programador1	9.360,00 €
+ Programador2	4.800,00 €

Figura 115 - Coste Total pro Recurso

## 10.2 - Recursos Materiales

Por otro lado, existen una serie de necesidades materiales, las cuales han de ser tenidas en cuenta a la hora de realizar la estimación económica del proyecto y que son los elementos físicos necesarios para completar el proyecto. Estos serán entregados al Cliente al terminar el mismo.

Dentro de este proyecto, el único recurso material con el que es necesario contar se trata del coste de compra/alquiler y utilización de servidor web en donde poder instalar la aplicación, y en donde poder alojar la información necesaria.

No se incurrirán en costes de licencias por compra de aplicaciones nuevas, dado que el desarrollo será realizado íntegramente con aplicaciones de código abierto.

Se ha optado por utilizar un servicio de Cloud Computing de cara a la elección del lugar en el que montar la aplicación, como es el Amazon Web Service.

Se ha elegido este tipo de servidor por varios motivos como son entre otros:

- No es necesario hacer una gran inversión anticipada, lo que podrá rebajar el coste directo del proyecto
- Coste de mantenimiento de servidor más bajo, dado que no es mantenido por la propia empresa
- Mayor velocidad y fiabilidad, dado que no trabajamos bajo un único servidor

Los requisitos que se propone son necesarios son los siguientes:

- 2 servidores
- Más de 200 GB de memoria RAM
- Acceso de manera independiente sobre los datos de la BBDD desde cualquier servidor
- Módulo de almacenaje de 500 TB de memoria.

Your AWS environment : EU (Ireland)					
Closest AWS Instances					
# Instances	Instance	vCPU	RAM (GiB)	Optimize by	Instance type
2	r3.xlarge	32	244	RAM	3 Yr. Heavy Utilization RI
0	t2.micro	1	1	RAM	3 Yr. Heavy Utilization RI

Storage (TB)		
SAN	NAS	Object
500	0	0

Bandwidth (Mbps)	
Pipe Size	Peak/Avg. Ratio
0	0

Figura 116 - Requisitos de servidor y disco

Con estos requisitos tendríamos una tarifa tri-anual de :

3 Yr. Total Cost of Ownership		
	On-Premises	AWS
Server	\$79,089	\$77,195
Storage	\$1,313,800	\$538,805
Network	\$6,599	\$ -
IT-Labor	\$ -	\$ -
Total	\$1,399,488	\$616,000

AWS cost includes business level support

Figura 117 - Tarifa tri-anual de coste de servidor

Aproximadamente unos 800 euros trianuales, primera cuota la cual será incluida al coste del proyecto.

## 10.3 - Gastos Indirectos

A parte de los gastos citados anteriormente, existen una serie de costes indirectos asociados, que son aquellos por los que la empresa debe ingresar a través de los proyectos que realice y en los cuales se incurre se realice o no proyecto. Son también llamados costes generales.

Dentro de esta categoría entran gastos como:

- Coste de alquiler y amortización de Inmueble
- Coste de alquiler y amortización de equipos informáticos
- Coste de servicios básicos como Luz, Agua, Red, etc
- Coste de Tasas ( I.A.E, tasas municipales)
- Coste de personal que no imputa directamente a proyecto, como pueden ser directivos, comerciales, recursos humanos, etc

Por lo normal, estos gastos son paliados mediante incrementos en el gasto de los proyectos que la empresa vende. Por parte de PRR Business Solution S.L, el incremento que han de sufrir los proyectos que la empresa lleve a cabo esta estimado en un 3%, que tendrá que ser sumado a la cifra final por la cual este sea vendido.

## 10.4 - Presupuesto Final

El presupuesto que finalmente será presentado al cliente, se basará en la suma de los gastos en los que incurrirá la empresa tanto de Recursos Humanos, materiales , así como un incremento que se aplica a cada uno de los proyectos de la empresa por posibles contingencias que no hayan sido estimadas, y gastos indirectos de la empresa.

Por lo tanto, el presupuesto final del proyecto seguirá la siguiente formula:

Presupuesto = (coste RR.HH + coste materiales)\*costes indirectos\*Contingencia\* IVA

De esta manera, y con lo visto en apartados anteriores, obtenemos el coste de recursos humanos que será aplicable al proyecto:

Resource Name	Work	Cost
+ Analista Funcional	480 hrs	9.120,00 €
+ Diseñador	432,73 hrs	6.490,91 €
+ Responsable de Calidad	26,67 hrs	373,33 €
+ Jefe de Proyecto	820 hrs	19.680,00 €
+ Programador1	936 hrs	9.360,00 €
+ Programador2	480 hrs	4.800,00 €

Figura 118 - Coste por Recurso

Lo que nos hace un total de 49.824,24€.

Por otro lado, a tenemos un coste de compra de servidor de 800 € que han de ser tenidos en cuenta.

Como hemos venido comentando con anterioridad, la empresa incurre diariamente en una serie de gastos indirectos, que han de ser recuperados mediante un sobrecoste de los proyectos de la empresa. Este está presupuestado actualmente como un 3% del coste de cada uno de los proyectos que se realicen.

Por último, estableceremos la contingencia sobre el gasto la cual actualmente está definida dentro de la empresa como un 8% del proyecto

De esta manera, el coste final del proyecto, y aplicando la formula anteriormente descrita será el que sigue:

Presupuesto Final Sin IVA= ((49.824,24€ + 800€) \*3%)\*8% = 56.314,40€

A lo que tendremos que aplicarle el 21% de IVA aplicable por lo que el resultado del presupuesto global del proyecto será tal y como se indica en la siguiente factura



## **PRESUPUESTO FINAL PROYECTO SISTEMA DE GESTIÓN DE CARTERA**

El presupuesto total de este proyecto asciende a la cantidad de SESENTA Y OCHO MIL, CIENTO CUARENTA CON CUARENTA Y DOS EUROS.

<b>68.140,42€</b>
-------------------

Figura 119 - Coste Total del Proyecto

Leganés a 21 de Enero de 2015

El ingeniero proyectista

Fdo. Pablo Rincón Rodríguez

## 11 - Conclusiones y líneas futuras

Durante la realización de un proyecto, y mediante todo su ciclo de vida, debido a muchas causas como son, estrés, tiempos ajustados, concentración en problemas que han surgido o tratando de evitar los mismos, se hace muy complicado realizar un estudio del proyecto, y ver cómo va avanzando y tomando forma el mismo a lo largo del tiempo.

Es por eso que a la finalización de todo proyecto es necesario echar la vista atrás y observar desde el inicio del mismo hasta el presente actual como ha ido avanzando, cuáles eran los requisitos iniciales del usuario, si han sido cumplidos, que problemas ha habido, que se ha ganado con el proyecto, etc.

Por otro lado, es necesario no quedarse en la entrega de la aplicación finalizada, la obtención de los distintos beneficios que este proporciona, etc, sino que hay que tratar de ver que otras posibilidades tiene la aplicación a futuro, a modo de poder generar nuevas propuestas al usuario final, para que este vea la necesidad de crear otro proyecto y de este modo la compañía continúe teniendo ingresos de este cliente.

A continuación se presentan las conclusiones principales en la realización de este proyecto y las posibles líneas futuras para su avance.

## 11.1 - Conclusiones

Una vez finalizado el proyecto, puesto en productivo y pasados los primeros días de funcionamiento, nos encontramos en disposición de ver si el resultado que se ha obtenido es el deseado, así como si los objetivos y requerimientos del usuario han sido cumplidos.

La base de la necesidad que el cliente ha ido transmitiendo desde el comienzo del proyecto se basaba en la necesidad de una herramienta para los empleados, y en la cual se pudieran llevar a cabo las tareas de comunicación internas más importantes, teniendo a todos los empleados actualizados en todo momento del estado de los distintos proyectos de la empresa, así como con capacidad de acceder a la distinta información del mismo. Todos estos puntos han sido cumplidos con satisfacción mediante la solución creada, en la cual toda la información se encuentra alojada dentro del mismo servidor, y con posibilidad de acceso por todos los usuarios.

Por otro lado, el usuario requería de algún método de gestión con el que este pudiera detectar posibles problemas de manera que estos pudieran ser solventados de la forma más efectiva posible. Mediante la posibilidad de creación de cualquier tipo de informe con los datos de la aplicación, y todas las funciones de planificación, exportación, servidores propios, etc, este objetivo ha sido cumplido de manera satisfactoria, otorgando al usuario mayor funcionalidad de la que el mismo solicitaba.

Por último, por parte del equipo de análisis, se propuso la posibilidad de crear automatizaciones de las asignaciones a proyectos, de cara a mejorar la eficacia de la empresa. El proceso automático funciona de forma correcta asegurando que siempre los desarrollos sean realizados por el mejor empleado posible.

La aplicación que finalmente ha sido desarrollada y presentada, ha buscado no solo la culminación de todos y cada uno de estos objetivos, sino que ha tratado de poner la primera piedra en la creación de una aplicación de gestión integral de la empresa, que ahorre costes y riesgos, y de esta manera aumente las ganancias.

Para obtener el resultado final que aquí se presenta, han sido muchas las dificultades que se han ido encontrado y que finalmente han podido ser solventadas. El primer inconveniente encontrado, fue el lenguaje en el que realizar la aplicación. No hubo duda en utilizar el lenguaje Java, ya que es el lenguaje con más proyección para las aplicaciones Web, pero este lenguaje no había sido estudiado ni aplicado hasta ahora. Tras los primeros problemas rápidamente se cogieron las ideas principales del mismo, y finalmente se consiguió realizar todas y cada una de las tareas deseadas, añadiendo de esta manera un gran conocimiento en este lenguaje, que sin duda podrá ser muy útil en el futuro.

Otro de los grandes inconvenientes se basa en la complejidad de utilización de librerías no creadas dentro del proyecto, como puede ser la visualización de datos en forma de árbol, exportaciones a Excel, tablas dinámicas, transferencias a servidores..., debido a la necesidad de cumplir requisitos muy rígidos para poder utilizarlos, teniendo en muchas ocasiones que realizar transformaciones de los datos, como puede ser el paso de los datos a arrays definidos, transformación a formato Json, etc.

Tras mucho estudio, finalmente se consiguió el objetivo, lo cual ha llevado a aprender mucho de cómo poder realizar la integración de componentes externos en una aplicación, y como poder aumentar el poder de una herramienta, mediante estos paquetes.

El último gran problema en la realización del proyecto, se encontró en la forma mediante la cual podrían compartirse los avances de la aplicación, para poder ser validadas. En un primer lugar se dotó a la aplicación capacidad de trabajar sobre base de datos Access, la cual es más fácil de intercambiarse, y el envío del documento war de despliegue junto con la base de datos se enviaba para su validación. Pero esta solución presentaba muchos problemas, como podía ser la necesidad de tener el validador de herramientas de despliegue de la aplicación, duplicidad de bases de datos, lo que en ocasiones provocaba fallos en un entorno, que no se daban en el otro, problemas en el intercambio de librerías, etc.

Finalmente para solventar este problema, y tras investigar mucho las distintas webs de servidores en los que se pueden subir aplicaciones web, y ver que prácticamente todas ellas no dan soporte a aplicaciones java, se optó por Amazon Web Service como servicio de "programación en la nube", donde finalmente se pudo encontrar un entorno al que subir el código, de manera que los avances del mismo pudiesen ser vistos al instante desde cualquier punto del mundo por el tutor, sin necesidad de realizar más acciones que acceder a una página web.

Prácticamente todos los proyectos, y este no iba a ser menos, ha ido creciendo con el paso del tiempo, incluyendo funcionalidad no planteada en un inicio, lo que ha hecho que tanto el análisis del proyecto como el diseño del mismo se ha ido adaptando y revisando de modo que finalmente todas las distintas partes del proyecto fuesen tomando forma, acabando teniendo una aplicación única en la cual todos sus componentes funcionen de una manera optima.

Para la realización de este proyecto han sido necesarias muchas horas de lectura y aprendizaje de distintas tecnologías, avances, búsqueda de distintas posibilidades para poder llevar a cabo los objetivos deseados, lo que en lo personal me ha llevado a un aumento muy amplio de mi conocimiento de muchas tecnologías que ni sabía prácticamente que existían, y muchos menos sabía cómo aplicarlas. Tras todas y cada una de los problemas, momentos en los que pensaba que no iba a ser posible finalmente acabo con el orgullo de haber sido capaz de dar forma a todas y cada una de las ideas que se han ido presentando y se pretendían hacer.

## 11.1 - Lineas Futuras

Como toda aplicación basada en tecnologías web, y debido al rápido avance de las aplicaciones, el aumento de las necesidades de las empresas, etc, esta no se puede decir nunca que esté acabada, dado que siempre caben mejoras y aumento de funcionalidad que se puede dar.

Este proyecto en particular tiene muchas vías de aumento y mejora, que han ido detectándose a lo largo del ciclo de vida del proyecto, y que han sido dejadas sin realizar para la realización de proyectos de ampliación.

En este apartado se van a proponer las ampliaciones o mejoras que resultan más interesantes a largo y corto plazo para el futuro de la aplicación Web desarrollada:

- ❖ Mejora del sistema de mensajería permitiendo a los propios usuarios crear mensajes enviarlos, así como la posibilidad de crear carpetas, para el almacenaje del correo, etc.
- ❖ Aumento de la capacidad de generación de informes, incluyendo la posibilidad de generar informes gráficos, como diagramas de barras, tartas, etc, que permitan ver la información de forma más grafica.
- ❖ Dado que las nuevas tecnologías van avanzando rápidamente, se plantea la posibilidad de la adaptación de la aplicación a entornos móviles como android y o IOS, de cara a poder tener acceso a la aplicación desde cualquier punto del mundo de una manera rápida y segura.
- ❖ Ampliación de la aplicación a gestiones internas del empleado, como puede ser, descarga de nóminas, inclusión del curriculum vitae, datos bancarios, etc.
- ❖ Ampliación de los servicios de alarmado , incluyendo mayor capacidad de tipologías de alarma, pudiendo aumentar los casos de actuación de la misma.

## 12 - Anexos

Se adjuntan los anexos de pruebas de sistema y manual de usuario que han sido generados como documentación adicional al proyecto:

### 12.1 - Anexo Pruebas Sistema Gestion Cartera

Dentro del anexo que se presenta a continuación, se adjuntan todas las pruebas realizadas al sistema para la comprobación de que los desarrollos realizados son correctos y cumplen con cada una de las pruebas.

El cuadro de pruebas de está formado por los siguientes elementos:

- Test Case --> Identificador único de la prueba
- Descripción --> Breve descripción de la prueba que ha de realizarse
- Precondiciones --> Condiciones que han de cumplirse para que la prueba pueda llevarse a cabo
- Resultado esperado --> Resultado que ha de darse para que la prueba pueda darse como Ok
- Resultado --> Resultado Final obtenido en la prueba

Toda la documentación de este anexo, está disponible dentro del documento "Anexo Pruebas Sistema Gestion Cartera.pdf"

### 12.2 - Anexo Manual de Usuario

Mediante el siguiente manual de usuario se pretende dar explicar de manera práctica y con imágenes cada una de las posibilidades que tiene el usuario sobre la aplicación, así como mostrar cómo sacar el mayor partido posible a todas y cada una de las funcionalidades de la misma.

Toda la documentación de este anexo, está disponible dentro del documento "Anexo Manual de Usuario Gestion Cartera.pdf"

## 12.3 - Anexo Presupuesto de Proyecto

Dentro del documento PresupuestoPFC-Gestión Cartera v1.0.pdf, podremos encontrar el presupuesto completo del proyecto, con el desglose de costes por concepto.

## 12.4 - Anexo preparación base de datos.

De cara al correcto funcionamiento dentro del entorno de producción de la funcionalidad de la herramienta, es necesario precargar los datos de las tablas de catalogo de la aplicación, mediante las siguiente sentencias de inserción:

### INSERCIONES TABLA GESTION OK

```
insert into Gestion_OK ('1','Las credenciales son correctas. Bienvenido al Sistema','Pagina_Inicio.html');
insert into Gestion_OK ('2','La password ha sido modificada correctamente','Empleados_Inicio.html');
insert into Gestion_OK ('3','El Usuario ha sido actualizado correctamente','Pagina_Inicio.html');
insert into Gestion_OK ('4','Se ha modificado correctamente la gestión de derechos','Pagina_Inicio.html');
insert into Gestion_OK ('5','El empleado ha sido dado de baja correctamente','Pagina_Inicio.html');
insert into Gestion_OK ('6','Los datos del empleado han sido modificados correctamente','Pagina_Inicio.html');
insert into Gestion_OK ('7','La Alarma ha sido creada correctamente','Pagina_Inicio.html');
insert into Gestion_OK ('8','La Alarma ha sido modificada correctamente','Pagina_Inicio.html');
insert into Gestion_OK ('9','La Alarma ha sido eliminada correctamente','Pagina_Inicio.html');
insert into Gestion_OK ('10','El informe ha sido correctamente planificado','Informes_Inicio.html');
insert into Gestion_OK ('11','El informe ha sido desplanificado correctamente','Informes_Inicio.html');
insert into Gestion_OK ('12','El Reporte ha sido almacenado correctamente','Reporte_Horas.html');
insert into Gestion_OK ('13','El Reporte ha sido almacenado correctamente','Pagina_Inicio.html');
insert into Gestion_OK ('14','La Solicitud ha sido correctamente registrada','Pagina_Inicio.html');
insert into Gestion_OK ('15','Asignacion Correcta','Pagina_Inicio.html');
insert into Gestion_OK ('16','El Proyecto ha sido cerrado con éxito','Pagina_Inicio.html');
insert into Gestion_OK ('17','El informe ha sido guardado como favorito','Pagina_Inicio.html');
insert into Gestion_OK ('18','La documentacion ha sido correctamente subida al servidor','Pagina_Inicio.html');
insert into Gestion_OK ('19','La documentacion ha sido correctamente descargada del servidor','Pagina_Inicio.html');
insert into Gestion_OK ('20','El informe ha sido exportado con exito','Pagina_Inicio.html');
insert into Gestion_OK ('21','El informe ha sido eliminado de la lista de favoritos','Pagina_Inicio.html');
insert into Gestion_OK ('22','El informe ha sido almacenado como favorito','Pagina_Inicio.html');
insert into Gestion_OK ('23','El proyecto ha sido eliminado con exito','Pagina_Inicio.html');
insert into Gestion_OK ('24','El proyecto ha sido modificado con exito','Pagina_Inicio.html');
insert into Gestion_OK ('25','El informe ha sido borrado con exito','Ver_Informes_Descarga.html');
insert into Gestion_OK ('26','El documento ha sido borrado correctamente del servidor','Cartera_Inicio.html');
```

### INSERCIONES TABLA GESTION ERRORES

```
insert into Gestion_Errores values ('1','No existe ningun usuario con los datos introducidos','index.jsp');
insert into Gestion_Errores values ('2','Ya existe un Empleado con este login','Pagina_Inicio.html');
insert into Gestion_Errores values ('3','Password antigua incorrecta','Pagina_Inicio.html');
insert into Gestion_Errores values ('4','La nueva password no coincide','Pagina_Inicio.html');
insert into Gestion_Errores values ('20','Error_Interno','Pagina_Inicio.html');
insert into Gestion_Errores values ('21','El proyecto a insertar ya existe','Pagina_Inicio.html');
insert into Gestion_Errores values ('22','El proyecto buscado no existe','Pagina_Inicio.html');
insert into Gestion_Errores values ('23','El conocimiento a insertar ya existe','Pagina_Inicio.html');
insert into Gestion_Errores values ('24','Este empleado ya tiene insertado este conocimiento','Alta_Empleado_2.html');
```

```

insert into Gestion_Errores values ('30','La alarma a insertar ya existe','Pagina_Inicio.html');
insert into Gestion_Errores values ('35','Error!! Todos los campos de la alarma han de estar informados','Aniadir_Alarma.jsp');
insert into Gestion_Errores values ('40','Este informe ya se encuentra planificado','Informes_Inicio.html');
insert into Gestion_Errores values ('100','Error!!No tiene acceso a esta funcionalidad.','Pagina_Inicio.html');
insert into Gestion_Errores values ('25','Error. La documentacion no ha podido ser subida al servidor','Upload_Ficheros.html');
insert into Gestion_Errores values ('26','No existe ningún documento asociado a este proyecto','Cartera_Inicio.html');
insert into Gestion_Errores values ('27','Error al descargar alguno de los documentos solicitados','Buscar_Proyecto_Lectura.html');
insert into Gestion_Errores values ('41','Error!! El informe no ha podido ser exportado.','Ver_Informe.jsp');
insert into Gestion_Errores values ('28','Error. El infomre no ha podido ser borrado del servidor','Informes_Inicio.html');
insert into Gestion_Errores values ('28','Error. La documentación no ha podido ser borrada del servidor','Informes_Inicio.html');
insert into Gestion_Errores values ('5','ERROR. El Reporte de horas ya existia previamente','Pagina_Inicio.html');
insert into Gestion_Errores values ('42','Error!! Ya existe un informe favorito con ese nombre.','Informes_Inicio.html');

```

#### INSERCIONES TABLA CAT\_CONDICIONES\_ALARMAS

```

insert into Cat_Condiciones_Alarmas values ('Ini Aceptacion','0');
insert into Cat_Condiciones_Alarmas values ('Ini Cierre','0');
insert into Cat_Condiciones_Alarmas values ('Ini Construccion','0');
insert into Cat_Condiciones_Alarmas values ('Ini Despliegue','0');
insert into Cat_Condiciones_Alarmas values ('Ini Macrodisenio','0');
insert into Cat_Condiciones_Alarmas values ('Ini Microdisenio','0');
insert into Cat_Condiciones_Alarmas values ('Ini Oferta','0');
insert into Cat_Condiciones_Alarmas values ('Ini Pruebas_de_Sistema','0');
insert into Cat_Condiciones_Alarmas values ('Ini Pruebas_Integradas','0');
insert into Cat_Condiciones_Alarmas values ('Fin Aceptacion','0');
insert into Cat_Condiciones_Alarmas values ('Fin Cierre','0');
insert into Cat_Condiciones_Alarmas values ('Fin Construccion','0');
insert into Cat_Condiciones_Alarmas values ('Fin Despliegue','0');
insert into Cat_Condiciones_Alarmas values ('Fin Macrodisenio','0');
insert into Cat_Condiciones_Alarmas values ('Fin Microdisenio','0');
insert into Cat_Condiciones_Alarmas values ('Fin Oferta','0');
insert into Cat_Condiciones_Alarmas values ('Fin Pruebas_de_Sistema','0');
insert into Cat_Condiciones_Alarmas values ('Fin Pruebas_Integradas','0');

```

#### INSERCIONES TABLA CAT\_CRUCES\_TABLAS

```

insert into Cat_Cruces_Tablas values ('Cat_Tipos_Alarmas','Alarmas','Cat_Tipos_Alarmas.tipo_alarma=Alarmas.tipo_aviso');
insert into Cat_Cruces_Tablas values ('Alarmas','Cat_Condiciones','Alarmas.campo_condicion=Cat_Condiciones.condicion');
insert into Cat_Cruces_Tablas values ('Empleados','Conocimientos_Empleados','Empleados.usuario=Conocimientos_Empleados.usuario');
insert into Cat_Cruces_Tablas values ('Mensajes_Sistema','Empleados','Mensajes_Sistema.usuario=Empleados.usuario');
insert into Cat_Cruces_Tablas values ('Empleados','Derechos','Empleados.perfil=Derechos.perfil');
insert into Cat_Cruces_Tablas values ('Login','Empleados','Login.usuario=Empleados.usuario');
insert into Cat_Cruces_Tablas values ('Empleados','Rel_Proyecto_Empleado','Empleados.usuario=Rel_Proyecto_Empleado.usuario');
insert into Cat_Cruces_Tablas values ('Empleados','Reporte_Horas','Empleados.usuario=Reporte_Horas.usuario');
insert into Cat_Cruces_Tablas values ('Empleados','Solicitudes','Empleados.usuario=Solicitudes.usuario');
insert into Cat_Cruces_Tablas values ('Empleados','Informes','Empleados.usuario=Informes.creador');
insert into Cat_Cruces_Tablas values ('Empleados','Cat_Equipos','Empleados.proyecto_asignado=Cat_Equipos.equipo');
insert into Cat_Cruces_Tablas values ('Rel_Proyecto_Empleado','Proyecto','Rel_Proyecto_Empleado.codigo_proyecto=Proyecto.codigo_proyecto');
insert into Cat_Cruces_Tablas values ('Proyecto','Planificacion_Proyecto','Proyecto.Codigo_proyecto=Planificacion_Proyecto.codigo_proyecto');
insert into Cat_Cruces_Tablas values ('Proyecto','Conocimientos_Proyecto','Proyecto.codigo_proyecto=Conocimientos_Proyecto.codigo_proyecto');
insert into Cat_Cruces_Tablas values ('Proyecto','Cat_Releases','Proyecto.release_objetivo=Cat_Releases.cod_release');
insert into Cat_Cruces_Tablas values ('Proyecto','Cat_Fases','Proyecto.fase=Cat_Fases.fase');
insert into Cat_Cruces_Tablas values ('Informes','Planificacion_Informes','Informes.nombre_informe=Planificacion_Informes.informe');

```



```
insert into Cat_Cruces_Tablas values
('Conocimientos_Empleados','Cat_Herramientas','Conocimientos_Empleados.Conocimiento=Cat_Herramientas.herramienta');
insert into Cat_Cruces_Tablas values ('Alarmas','Empleados','Alarmas.perfil_afectado=Empleados.perfil');
```

#### INSERCIONES TABLA CAT EQUIPOS

```
insert into Cat_Equipos values ('Data Warehouse','2014-07-30','0','princon3');
insert into Cat_Equipos values ('TIBCO','2014-07-30','0','jllopez');
insert into Cat_Equipos values ('ARBOR','2014-07-30','0','princon3');
insert into Cat_Equipos values ('Clarify','2014-07-30','0','fpaniag');
insert into Cat_Equipos values ('NewCrm','2014-07-30','0','princon3');
insert into Cat_Equipos values ('IRIS','2014-07-30','0','princon3');
insert into Cat_Equipos values ('Chordiant','2014-07-30','0','princon3');
insert into Cat_Equipos values ('SAP','2014-07-30','0','princon3');
insert into Cat_Equipos values ('RRHH','2014-08-16','0','princon3');
```

#### INSERCIONES TABLA CAT HERRAMIENTAS

```
insert into Cat_Herramientas values ('Java','0');
insert into Cat_Herramientas values ('C','0');
insert into Cat_Herramientas values ('Pascal','0');
insert into Cat_Herramientas values ('C++','0');
insert into Cat_Herramientas values ('MySQL','0');
insert into Cat_Herramientas values ('Oracle','0');
insert into Cat_Herramientas values ('Teradata','0');
insert into Cat_Herramientas values ('Sybase','0');
```

#### INSERCIONES TABLA CAT RELEASES

```
insert into Cat_Releases values ('ES1301RE','0');
insert into Cat_Releases values ('ES1303RE','0');
insert into Cat_Releases values ('ES1305RE','0');
insert into Cat_Releases values ('ES1307RE','0');
insert into Cat_Releases values ('ES1309RE','0');
insert into Cat_Releases values ('ES1311RE','0');
insert into Cat_Releases values ('ES1401RE','0');
insert into Cat_Releases values ('ES1403RE','0');
insert into Cat_Releases values ('ES1405RE','0');
insert into Cat_Releases values ('ES1407RE','0');
insert into Cat_Releases values ('ES1409RE','0');
insert into Cat_Releases values ('ES1411RE','0');
insert into Cat_Releases values ('ES1501RE','0');
```

#### INSERCIONES TABLA CAT TIPOS ALARMAS

```
insert into Cat_Tipos_Alarmas values ('Aviso Responsable','0');
insert into Cat_Tipos_Alarmas values ('Correo','0');
insert into Cat_Tipos_Alarmas values ('Mensaje en Sistema','0');
```

#### INSERCIONES TABLA FUNCIONALIDAD PAGINA

```
insert into Funcionalidad_Pagina values ('Alta_Empleado.jsp','Adm_Alta_Empleado');
insert into Funcionalidad_Pagina values ('Aniadir_Alarma.jsp','Alar_Crear_Alarma');
insert into Funcionalidad_Pagina values ('Buscar_Alarma.jsp','Alar_Modificar_Alarma');
insert into Funcionalidad_Pagina values ('Buscar_Alarma_Eliminar.jsp','Alar_Borrar_Alarma');
insert into Funcionalidad_Pagina values ('Buscar_Empleado.jsp','Adm_Modificar_Datos_Empleado');
insert into Funcionalidad_Pagina values ('Buscar_Empleado_Eliminar.jsp','Adm_Dar_Baja_Empleado');
insert into Funcionalidad_Pagina values ('Cambiar_Derechos_Inicio.html','Adm_Cambiar_Derechos');
insert into Funcionalidad_Pagina values ('Crear_Solicitud.html','Emp_Solicitudes');
insert into Funcionalidad_Pagina values ('Informes_Favoritos.jsp','Inf_Informes_Favoritos');
```

```
insert into Funcionalidad_Pagina values ('Mostrar_Alarmas.html','Alar_Mostrar_Alarma');
insert into Funcionalidad_Pagina values ('Nuevo_Informe.html','Inf_Crear_Nuevo_Informe');
insert into Funcionalidad_Pagina values ('Nuevo_Proyecto.jsp','Car_Nuevo_Proyecto');
insert into Funcionalidad_Pagina values ('Planificar.jsp','Inf_Planificar_Informes');
insert into Funcionalidad_Pagina values ('Reporte_Horas.html','Emp_Reporte_Horas');
insert into Funcionalidad_Pagina values ('Ver_Datos_Propios.jsp','Emp_Ver_Datos_Propios');
insert into Funcionalidad_Pagina values ('Ver_Planificacion.jsp','Inf_Ver_Planificacion');
insert into Funcionalidad_Pagina values ('Ver_Proj_Asignados.html','Emp_Proyectos_Asignados');
insert into Funcionalidad_Pagina values ('Ver_Solicitudes.html','Emp_Solicitudes');
insert into Funcionalidad_Pagina values ('Ver_Solicitudes_Aprobar.html','Emp_Aprobar_Solicitudes');
insert into Funcionalidad_Pagina values ('Buscar_Proyecto.jsp-accion=modificar','Car_Modificar_Proyecto');
insert into Funcionalidad_Pagina values ('Buscar_Proyecto.jsp-accion=cerrar','Car_Cerrar_Proyecto');
insert into Funcionalidad_Pagina values ('Buscar_Proyecto.jsp-accion=eliminar','Car_Eliminar_Proyecto');
insert into Funcionalidad_Pagina values ('Buscar_Proyecto.jsp-accion=lectura','Car_Buscar_Proyecto');
```

## 13 - Bibliografía

A continuación se pasa a detallar al material principal que ha sido consultado a lo largo del proyecto de cara a adquirir el conocimiento necesario para poder llevarlo a cabo.

Se detalla tanto el material escrito, como el material principal consultado online.

### 13.1 - Libros

1. **García Pérez, Fernando - Chamorro Atance, Félix - Molina López, José Manuel** - Informática de Gestión y Sistemas de Información. ( McGrawHill, 2000).
2. **Gutiérrez Gallardo, Juan Diego** - Guía práctica de MySQL ( Anaya Multimedia, 2004).
3. **García Sánchez, Álvaro** - Despliegue de Aplicaciones Web (GARCETA GRUPO EDITORIAL, 2012).
4. **Díez de Castro, Emilio** - INFORMATICA APLICADA A LA PLANIFICACION Y GESTION DE LA EMPRESA ( Pirámide, 2008).
5. **Pérez Montes, Francisco** . APRENDE A PROGRAMAR CON JAVA. (EDICIONES PARANINFO, S.A., 2012).
6. **Sawyer Mcfarland, David** - JAVASCRIPT Y JQUERY (ANAYA MULTIMEDIA, 2012).
7. **M.Marquez . Francisco** - UNIX: PROGRAMACION AVANZADA (3ª ED.) ( RA-MA, 2004).

## 13.1 - Páginas Web

1. **W3C**. Guía breve de los Servicios Web. Accedido en Enero de 2012.  
<http://www.w3c.es/Divulgacion/GuiasBreves/ServiciosWeb>.
2. **W3C**. Guía Breve sobre estándares Web. Accedido en Enero de 2012.  
<http://www.w3c.es/Divulgacion/Guiasbreves/Estandares>.
3. **Pozo, Juan R.** Traducción de la Especificación HTML 4.01 al castellano. Accedido en Abril de 2012. <http://html.conclase.net/w3c/html401-es/progreso.html>.
4. **W3C Consortium**. A graphical depiction of a very simple css document. Accedido en Agosto de 2012. <http://gaffney3.com/2008/04/08/css-naked-day/>.
5. **Álvarez Miguel A.** La tecnología Java para la creación de páginas Web con programación en el servidor. Accedido en Abril de 2012.  
<http://www.desarrolloweb.com/articulos/831.php>.
6. **SUN**. Conozca más sobre la tecnología Java Accedido en Abril de 2012.  
<http://www.java.com/es/about/>.
7. **Amazon EC2, Apache, MySQL, PHP, and Tomcat 7 Tutorial for Laura**. Accedido en Julio de 2014. <http://www.youtube.com/watch?v=pZFSSVVdqmw>.
8. **Guía de utilización de Javadoc**. Accedido en Mayo de 2014.  
<http://java-white-box.blogspot.com.ar/2012/08/javadoc-que-es-el-javadoc-como-utilizar.html>.
9. **Ejemplos de utilización de JQuery**. Accedido en Septiembre de 2013.  
<http://www.jqueryrain.com/demo/jquery-treeview/>.
10. **Ejemplos de utilización de datatables**. Accedido en Noviembre de 2013.  
<http://www.datatables.net/>.
11. **Manual de utilización de JSON**. Accedido en Septiembre de 2013.  
<http://www.w3schools.com/json/>.

12. **Guía de subida de fichero a servidor** Accedido en Julio de 2014.  
<http://notodojava.tredoo.com/subir-fichero-java-java-file-upload/>.
13. **Guía de utilización de Excel con Java.** Accedido en Marzo de 2014.  
<http://hablandojava.wordpress.com/2013/04/09/leer-y-escribir-en-ficheros-excel-desde-java-con-jakarta-poi/>.
14. **Guía de conexión de Java con Mysql.** Accedido en Septiembre de 2012.  
<http://todoenjava.blogspot.com.es/2013/02/conexion-base-de-datos-mysql-con-jdbc.html>.
15. **Guía de Javascript.** Accedido en Febrero de 2014.  
<http://www.manualdejavascript.com/section/manualdejavascript/>.

